University of the Philippines Manila
College of Arts and Sciences
Department of Physical Sciences and Mathematics

CSCoRE: Computer Science Collaborative Research Environment

A special problem in partial fulfillment
of the requirements for the degree of
Bachelor of Science in Computer Science

Submitted by:

Joanna Marielle P. Jimenez

April 2012

ACCEPTANCE SHEET

The Special Problem entitled "CSCoRE: Computer Science Collaborative Research Environment" prepared and submitted by Joanna Marielle P. Jimenez in partial fulfillment of the requirements for the degree of Bachelor of Science in Computer Science has been examined and is recommended for acceptance.

____

Richard Bryann L. Chua, M.Sc.
Adviser

EXAMINERS:

	Approved	**Disapproved**
1. Gregorio B. Baes, Ph.D. (*candidate*)	____________	____________
2. Avegail D. Carpio, M.Sc.	____________	____________
3. Aldrich Colin K. Co, M.Sc. (*candidate*)	____________	____________
4. Ma. Sheila A. Magboo, M.Sc.	____________	____________
5. Vincent Peter C. Magboo, M.D., M.Sc.	____________	____________
6. Geoffrey A. Solano, M.Sc.	____________	____________
7. Bernie B. Terrado, M.Sc. (*candidate*)	____________	____________

Accepted and approved as partial fulfillment of the requirements for the degree of Bachelor of Science in Computer Science.

____

Avegail D. Carpio, M.Sc.
Unit Head
Mathematical and Computing Sciences Unit
Department of Physical Sciences
and Mathematics

____

Marcelina B. Lirazan, Ph.D.
Chair
Department of Physical Sciences
and Mathematics

____

Reynaldo H. Imperial, Ph.D.
Dean
College of Arts and Sciences

Abstract

Generally, collaboration is the act of working together to achieve a common goal. Its scope does not only include the idea of working together but also how the participants can work together. With the increasing availability of the Internet, companies are enabled to develop cost-effective collaborative solutions called collaboratories. A collaboratory is a virtual research environment that enables researchers based in different locations to work together and share their knowledge and facilities, thus enriching and speeding up both national and international research. Today, numerous collaboratories are created for different disciplines. However in the field of Computer Science, the only closest idea we have to collaboratories is the distributed software development. Though the existing technical infrastructure seems to provide adequate support for distributed software development, projects are most often undermined by poor software development environments and tools. Not only the diverse set of tools is reported to cause problems but also the different versions of a single tool potentially caused problems as well.

The Computer Science Collaborative Research Environment (CSCoRE) has been built to address these problems. CSCoRE is a tool for collaborative software development and software project management that is specifically designed for Computer Science research. It provides a common interface where participants can create and join projects and have a single storage for data and information. Furthermore, CSCoRE enables the members of each project to manage and share data, references, tools and other relevant materials among themselves. Also, the system helps in maintaining current and historical versions of the files of the project with the use of subversion.

Keywords: Collaboratory, Software Development System, Project Management System, Subversion

Contents

List of Figures

List of Tables

I. Introduction

A. Background of the Study

Generally, collaboration is the act of working together to achieve a common goal [1]. Currently, collaboration has gained popularity and is applied in various disciplines. Research collaboration has been growing over the last 20 to 30 years [1] due to the several benefits it provides. First, collaboration ensures a more effective use of the participants' talents. Second, collaboration is one way of transferring new knowledge, especially the tacit ones. Third, it may become a source of stimulation and creativity by bringing about clash of views and cross-fertilization of ideas. Fourth, it provides intellectual companion. Fifth, it enhances the potential visibility of work [2].

In the field of Computer Science, the wide-spread information communication technology enables a flexible form of collaboration [3]. Software development is one real-world situation where work regularly happens in a distributed fashion. Open-source software (OSS) is particular in this regard. These development projects have numerous programmers from many different parts of the world, who rarely meet face-to-face [4]. The distributed software development enables software production to take place independently of the geographical location of the individuals/organizations concerned [5]. Despite the difficulties imposed by distance, many OSS projects have managed to produce large, complex, and successful software systems which includes the Apache Web server [1] , Mozilla Web browser [2], and Linux kernel [3].

Nevertheless, the scope of collaboration does not only include the idea of working together but also how the participants can work together. Collaboration also consists of providing all individuals involved with suitable tools to get them work as a whole, making personal and group goals merge for an ultimate aim or task [6].

[1] http://www.apache.org

[2] http://www.mozilla.org

[3] http://www.linux.com/

Today, for Computer Science professionals, several tools are available to be used for collaboration: e-mails and online messaging for communication, file repositories for file and data sharing, forums for topic discussion, Software Configuration Management tools for effective handling of source codes and project management tools for project planning, tracking and maintenance.

B. Statement of the Problem

Even though tools which may promote successful collaboration are available, these tools are used and employed separately. Thus, the participants need to check and update each of the tools to keep everything synchronized. The only closest idea to collaboratories is the distributed software development.

While the existing technical infrastructure seems to provide adequate support for distributed software development, it is indicated that the development environment is not yet mature. Projects are most often undermined by poor software development environments and tools. Not only were a diverse set of tools reported to cause problems but also the different versions of a single tool potentially caused problems as well[5]. Moreover, applications which are built for distributed software development focus only on one specific area like the project management or file repository while other aspects like issue tracking, tools and reference management are still not integrated.

C. Objectives of the Study

1. To adapt the generic collaborative component of the Collaboratory for Epidempiological Research (CEpiR) to create the Computer Science Collaborative Research Environment (CSCoRe) which has 6 access levels:

   (a) System administrator

   (b) Project administrator

   (c) Project contributor

(d) Restricted user

(e) Registered user

(f) Non-registered user

2. CSCoRe has 8 modules:

   (a) Version control system

   (b) User management

   (c) References

   (d) Tools

   (e) Forums

   (f) Miscellaneous files

   (g) Issue tracking

   (h) Project management

3. The users have the following functionalities listed based on their scope of access

   (a) System-wide level includes the users whose functionalities are applicable on the whole system

      i. Allows the non-registered user to

         A. Request for an account

         B. Browse the list of projects in the system

      ii. Allows the registered user to

         A. Request for project creation

         B. Request for membership in a project

      iii. Allows the system administrator to

         A. Manage user accounts

         B. Manage the list of projects

(b) Project-wide level includes the users whose functionalities are applicable only in a specific project. Under this level is the project administrator and the system allows him to:

   i. Edit project information

   ii. Manage project members and their levels within the project and within each module

   iii. Terminate a project

(c) Module-wide level includes the users whose functionalities are applicable only in a module in a specific project

   i. Allows the restricted user to

      A. Download miscellaneous files

      B. Download references

      C. View the BibTex format of a reference

      D. Browse and check-out the files in the subversion repository

      E. Download tools

      F. Receive updates from the project

   ii. Allows the project contributor to

      A. Perform all the functions of a restricted user

      B. Upload miscellaneous files

      C. Upload references with details

      D. Make a comment to an uploaded reference

      E. Commit changes in the files in the subversion repository

      F. Upload tools

      G. Participate in the forum

D. Significance of the Project

This system promotes effective collaboration by integrating the essential tools in one application. The participants need not to have an account in different systems and keep each updated. Effective handling of documents which are needed throughout a project's lifetime is supported too. All files are kept in one application thus providing easy and quick access to documents, references and other files. Moreover, researchers have better collaboration even if they are from different geographical locations by setting up a common storage and interface for them to use. With the use of projects, the participants are able to share ideas and work together privately. Also, with the integration of workflow, the issue tracking and project management module can be customized according to how the participants want to do things.

E. Scope and Limitations

1. The issue tracking and project management modules are be based on a workflow integrated from another project
2. The system is not responsible for legality and appropriateness of the materials uploaded.
3. The system administrator is responsible for the legitimacy of the projects done within the system.
4. Any type of file can be uploaded in the references and miscellaneous files section.

F. Assumptions

1. The system administrator is trusted that he will not change his access level into a project administrator in order to see the contents of a project.
2. In the module-wide level, the project administrator may assume the role of project contributor.
3. All modules are available to all projects.
4. Checking of credentials of the user for account creation is manually done.

5. Mechanics of project creation is manually done.

II. Review of Related Literature

The rapid evolution of our data communications infrastructure is making distributed projects increasingly viable. Without a common infrastructure, computer-supported collaborative tools for distributed teams have been prohibitively expensive to build and maintain. However, the increasing availability of the Internet is enabling companies to develop cost-effective collaborative solutions in different fields [7].

With this, numerous collaboratories are created. To distinguish one from another, a seven category taxonomy was described by Bos, et al. and the groupings were mainly based on the resource and activity (see Table 1). In general, the level of management and sustainability of collaboratories get more difficult from the top left to the bottom right and that over time, the collaboratories move along the dimensions in both directions [8].

	Instrument (Tools)	Information (Data)	Knowledge (New Findings)
Aggregating • Across Distance • Loose Coupling • Often Asynchronously	Shared Instrument	Community Data System	Virtual Learning Community, Virtual Community of Practice
Co-creating • Across Distance • Tighter Coupling • Often Synchronously	Infrastructure	Open Contribution System	Distributed Research Center

Table 1: Collaboratory types by resource and activity

BioCoRE or Biological Collaborative Environment is an example of collaborative work environment. It is specialized for biomedical research, research management and training. Initial design work on BioCoRE has led to a web-based architecture focused on four primary interface paradigms: a workbench allows diverse computational tools to be applied to the problem at hand in a consistent manner, a notebook automates recording of research activities, electronic conferences held with collaborators can be saved and replayed, and multi-author documents can be prepared in a cross platform revision control system [9].

Sunderam et al., on the otehr hand, has designed the Collaborative Computing Frameworks (CCF). CCF is a suite of software systems, communications protocols, and tools that enable collaborative, computer-based cooperative work. CCF constructs a virtual work environment on multiple computer systems connected over the Internet, to form a Collaboratory. In this setting, participants interact with each other, simultaneously access and operate computer applications, refer to global data repositories or archives, collectively createand manipulate documents or other artifacts, perform computational transformations, and conduct a number of other activities via telepresence. The primary target domain for CCF is collaborative research in chemistry, physics, and biochemistry, and closely related areas. Research issues addressed in this project include problem solving environments and methodologies for laboratory and instrument-based scientific disciplines, and computer science issues in heterogeneous distributed systems [10].

Another example is the DISCOVER Web-based computational collaboratory. Its primary goal is to bring large distributed simulations to the scientists'/engineers' desktop by providing collaborative Web-based portals for monitoring, interaction and control. DISCOVER supports a 3-tier architecture composed of detachable thin-clients at the front-end, a network of interaction servers in the middle, and a control network of sensors, actuators, and interaction agents at the back-end [11].

In addition, due to the need of access to multiple, distributed data sources and analytic tools in many in silico investigations in bioinformatics, the BioExtract Server[4] is made. The BioExtract Server is a Web-based data integration applica-

[4] http://bioextract.org

tion designed to consolidate, analyze, and serve data from heterogeneous biomolecular databases in the form of a mash-up. The basic operations of the BioExtract Server allow researchers, via their Web browsers, to specify data sources, flexibly query data sources, apply analytic tools, download result sets, and store query results for later reuse. As a researcher works with the system, their steps are saved in the background [12].

Collaborative environments for computer science research has also been gaining interest now a days since most software development today is a team activity. Project team members collaboratively work on the tasks necessary to accomplish the various project milestones. Collaboration in software development is an open and very active research field. Motivated by the not few advantages of collaboration, many researchers have proposed collaboration tools to facilitate the process [13].

Gao et al. has implemented the research collaboratory for Open Source Software related research. The collaboratory includes a repository for the research-related information and a community for the researchers to cooperate with each other in the research. The collaboratory provides a method to share information, publish the research results, discuss it with peer researchers and promote future research in the domain [14].

The ability to manage the evolution of logical structures of software artifacts during design and implementation process is also crucial to the development of high-quality software [15]. Tools are also made to effectively handle those artifacts.

Fokaefs et al. has built a collaboration tool, the WikiDev 2.0 which adopts a wiki as the central platform in which to integrate information about the various artifacts of interest and to present views on this information that cut across the individual tool boundaries[13].

Mukherjee et al. on the other hand, has implemented a purely peer-to-peer environment and presents a peer-to-peer based version control system. Currently their system offers a CVS-like functionality to track the evolution of single files

[16].

Lam and Maheshwari has integrated project management tool in their system in order to facilitate effectively the process of project develeopment. The Distributed Software Project Management Tool (DSPM) is a distributed software project management information system delivering a collection of project management tools through a single interface. The collaboration of these tools gather, analyze, integrate and disseminate the outputs of various project management processes. The first prototype, designed by Surjaputra and Maheshwari, manages documents of the project through the lifetime of a software project. The second prototype is designed to manage, guide and monitor the full progress of the software project [17].

Ly has described in his paper a project management tool but added a feature also to promote collaboration. His work, Distributed Java Applet For Project Management On The Web, utilizes Java web-based application to create an online project management tool. One unique characteristics of this application which distinguishes it from other project management tools is that the server he used contains a notification mechanism for distributing information to users in real time, enabling the distributed project team to have up-to-date views of project plans and to-do lists. It also support the concept of a collaborative document in which multiple users can make edits to the same document simultaneously [7].

Online colloaborative development environments have inceasing popularity too especially in developing Open Source Software. The most well-known today is the SourceForge [5], from CollabNet, with over 311,000 hosted project. The original mission of SourceForge was to enrich the open source community by providing a centralized place for developers to control and manage OSS projects. SourceForge offers a variety of free services: web interface for project administration, space for web content and scripts, trackers (for reporting bugs, submitting support requests or patches to review, and posting feature requests), mailing lists and discussion

[5] http://sourceforge.net

forums, download notification of new releases, shell functions and compile farm, and CVS-based as well as Subversion based configuration management [18].

Another online collaborative development environments is Trac [6] which provides an integrated wiki, an issue tracking system and a front-end interface to SCM tools, usually Subversion. Project overview and progress tracking is allowed by setting a roadmap of milestones, which include a set of so-called tickets (i.e., tasks, feature requests, bug reports and support issues), and by viewing the timeline of changes. Trac also allows team members to be notified about project events and ticket changes through email messages and RSS feeds [18].

[6] http://trac.edgewall.org

III. Theoretical Framework

A. Collaboratory

According to the SURF Foundation[7], a collaboratory is a virtual research environment that enables researchers based in different locations to work together and share their knowledge and facilities, thus enriching and speeding up both national and international research. William Wulf coined the term collaboratory by merging the the words collaboration and laboratory. He defined the word as "center without walls" [19].

Generally, a collaboratory assists scientific interaction within a team by creating a new, artificial environment in which individuals can interact. It basically connects people, information and facilities as illustrated in Figure 1. In order to connect people with one another, a collaboratory makes use of different communication and groupware services. Information are provided within reach through the digital libraries, electronic publications, file sharing facilities and data repositories. On the other hand, several facilities are also made available with the widespread use of internet and online services.

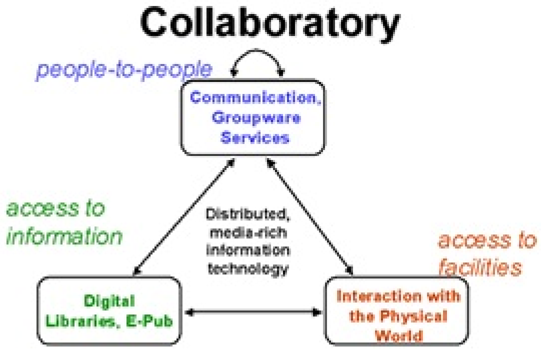

Collaboratory
people-to-people
Communication, Groupware Services
access to information
Distributed, media-rich information technology
access to facilities
Digital Libraries, E-Pub
Interaction with the Physical World

Figure 1: Collaboratory Triangle

To facilitate scientific work, collaboratory systems must support the sharing of secure data, analysis, instruments, and interaction spaces. Current implementations use an integrated set of crossplatform tools such as electronic notebooks, videoconferencing systems, electronic whiteboards, shared screens, information-

[7] http://www.surffoundation.nl

access tools, and instrument-control tools. Figure 2 illustrates how different tools provide varying functionality in interactions depending upon the static or dynamic nature of the information exchange as well as upon the synchronous or asynchronous nature of the session [19].

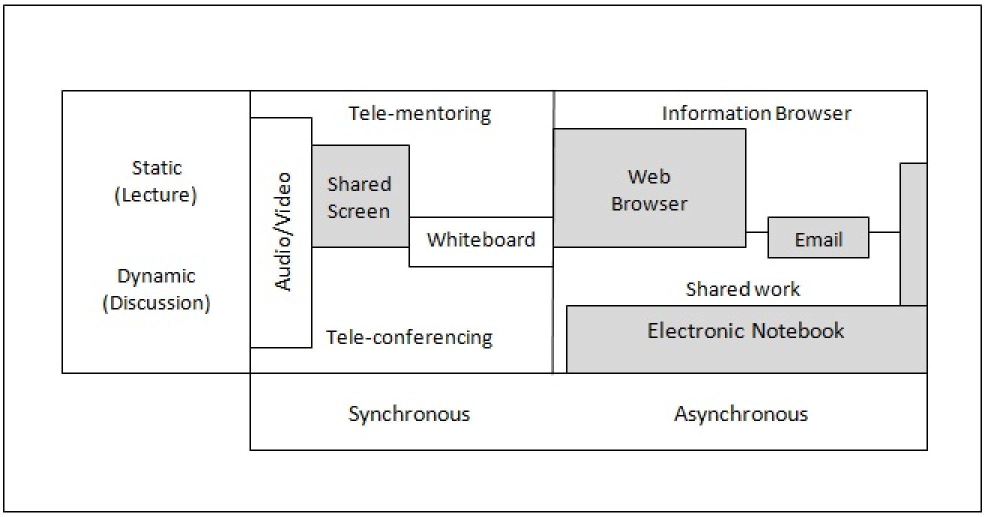


Figure 2: General collaboratory tools

E-mail supports collaboration via a time-serial dialog. Videoconferencing supports real-time discussion and, with the addition of graphics and whiteboard capabilities, presentation and brainstorming. Electronic notebooks provide distributed access to data, as well as automated data entry, searching, and other information processing not possible in a paper notebook.

Because the collaboratory concept brings all the scientific resources used by researchers into the mix, both realtime work and asynchronous collaboration are possible. The effect of having all scientific resources available to all researchers moves a remote collaborator from the role of part-time consultant to coworker.

B. Collaboratory for Epidemiological Research (CEpiR)

The Collaboratory for Epidemiological Research is a system built in order to allow Philippine-based researchers and epidemiologists to collaborate and share data and information among themselves. It is done by providing an interactive system where researchers and epidemiologists can join projects and have a common storage for data and information.

CEpiR is built using the Model View Presenter (MVP) framework which is the framework highly recommended by Google to use with GWT. In MVP, the Model represents the business objects of the system. The View represents all UI components while the Presenter contains all the logic for the system. A fourth component, known as the AppController, controls all the logic outside the Presenter like for example, history management and view transition logic. The Presenter catches all the events done in the View and passes the event to the AppController if for example the history is changed. Both the Presenter and AppController calls the RPC service if access to the database is needed[20].

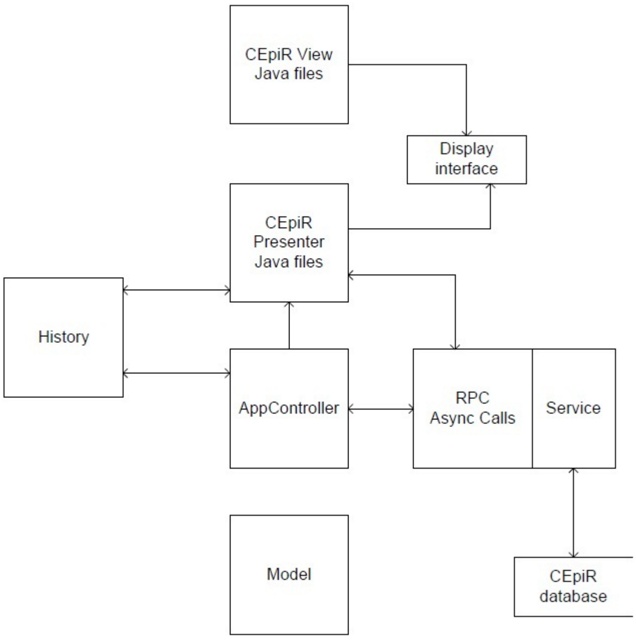


Figure 3: CEpiR Archtecture

C. Collaborative Computing Frameworks (CCF)

Collaborative Computing Frameworks[8] is a suite of software systems, communications protocols, and tools that enable collaborative, computer-based cooperative work. A schematic of the CCF software architecture is shown in Figure 4.

[8] http://www.mathcs.emory.edu/ccf/

CCF operates over standard Internet protocols or over high speed local networks, but builds its own fast and efficient multiway protocols thereupon - these core modules are identified as Collaborative Computing Transport Layer (CCTL) and CCF-API in the diagram. The data management and computing management modules provide infrastructural support for these functions, and the CCSM module is responsible for creating and managing the shared workspace. CCFX is an auxiliary subsystem that is used to multiplex CCF-unaware applications. Tools and CCF-aware applications interface to the different core modules as appropriate [10].

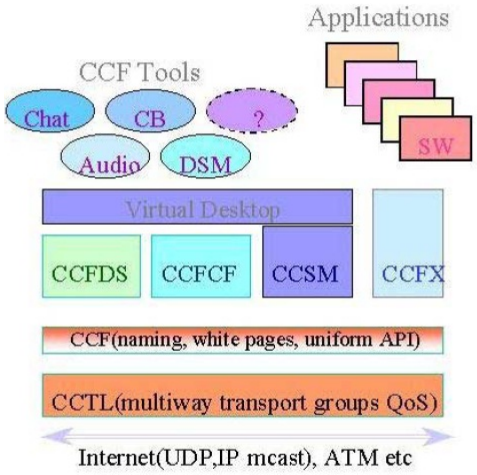

Applications
CCF Tools
Chat
CB
?
Audio
DSM
SW
Virtual Desktop
CCFDS
CCFCF
CCSM
CCFX
CCF(naming, white pages, uniform API)
CCTL(multiway transport groups QoS)
Internet(UDP,IP mcast), ATM etc

Figure 4: CCF Architecture

D. SourceForge Research Data Archive (SRDA)

The SourceForge Research Data Archive[9] is an online research collaboratory built to facilitate Open Source Software (OSS) related research. It also serves as a repository of monthly database dumps from SourceForge[10].

The collaboratory is built using a three-tier hierarchy as shown in Figure 5. The presentation tier is the user interface of the research collaboratory. It is responsible for formatting the information and delivering it to the end user. The major tasks of this tier includes providing various access methods to the backend

[9] http://zerlot.cse.nd.edu

[10] http://sourceforge.net

repository and providing community support for the researchers. All users, including anonymous users, can access the community; registered users are allowed to post on the community and only specially authorized users can access the back-end repository through the web interface or programmable interface. For the public community service, wiki technology is used.

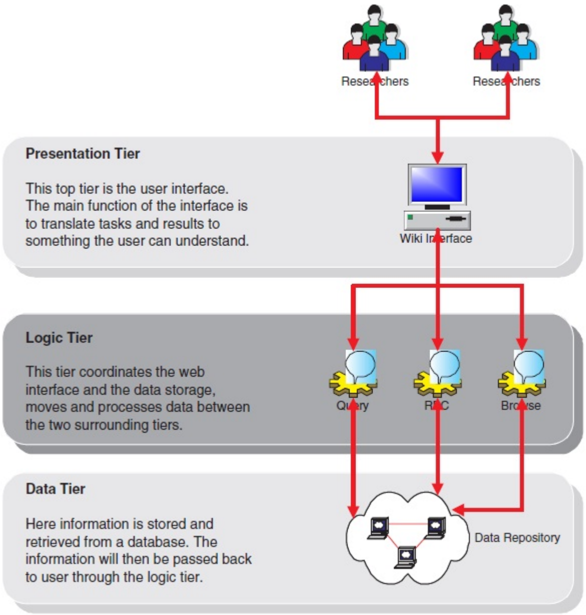

Researchers
Researchers
Presentation Tier
This top tier is the user interface. The main function of the interface is to translate tasks and results to something the user can understand.
Wiki Interface
Logic Tier
This tier coordinates the web interface and the data storage, moves and processes data between the two surrounding tiers.
Query
RPC
Browse
Data Tier
Here information is stored and retrieved from a database. The information will then be passed back to user through the logic tier.
Data Repository

Figure 5: OSS Architecture

The next tier is the logic tier which implements most of the processes and functions provided in the presentation tier. Usually these processes and functions are based on the information stored in the data tier. There are two kinds of access methods to the back-end repository provided in the collaboratory: web access and programmable access. Web access is the simple method to access the back-end repository and it is used to implement the database query interface. Authorized users can submit queries through a provided web page then the result is saved in the user directory and be made available for download by the user. Files are used instead of returning the result directly to the user to avoid unexpected network delays and other problems. The programmable access, on the other hand, is provided to allow the user to create sophisticated requests that cannot be described using single query or use procedures in the query.

The data tier includes a data repository at the back end of the system to store all the related information. The major tasks of this tier include an appropriate schema, an appropriate backup strategy and an appropriate load-balancing technique. In the database, there are multiple schemas. Every schema represents a single month, and stores the database dump from SourceForge of the corresponding month. Since the repository should expand only every month, a new database dump is restored. A monthly full repository backup is good enough to keep the repository fully recoverable. As to load balancing, connection management is applied to relieve the back-end server from a possible heavy load of queries [14].

E. Distributed Software Project Management Tool (DSPM-Tool)

The Distributed Software Project Management Tool is an integration of tools opening the realms of software project management to users distributed across the world. The collaboration of these tools gather, analyse, integrate and disseminate the outputs of various project management processes. The first prototype presents the core of a software repository and configuration management. It manages documents of the project through the lifetime of a software project The second prototype integrates the concepts of task and team management to improve the quality of software projects.

The DSPMtool is a client/server system which uses an ActiveX Data Object (ADO) to provide an abstract interface to the physical database implementation. The Visual Basic environment provides ADO and an Object Link Embedded (OLE) database model. The protocol defined between the server and the client operates on a Transmission Control Protocol/Internet Protocol (TCP/IP) backbone. It also uses popular object middlewares, such as the DCOM and ActiveX technology. The Distributed Component Object Model (DCOM) is a component model or a software architecture allowing the DSPMtool to be built from binary software components distributed across the network. The ActiveX controls are among the many types of components that use the DCOM component model to deliver particular sets of features used in the DSPMtool application. Many of the graphical interface and system functionality in the DSPMtool are provided by various ActiveX controls and other DCOM based components [17].

F. ActionPlan

Netmosphere ActionPlan is a Web-enabled project management tool for distributed teams. It is consisted of two Java applets, ActionPlan and the HotSheet, and a collaboration server also implemented in Java.

The ActionPlan applet allows project managers to create project plans and assign resources within a workgroup. As changes occur throughout the project, project managers can apply task changes to the plan that are communicated in real time to the rest of the team.

The HotSheet applet serves as an automatically updating to-do list for individual contributors. Tasks can be checked off as they are completed. HotSheet contents are tailored to the specific user and show only those tasks on all projects assigned to the logged-in user.

The Java Application Server (JAS) is the underlying server architecture which maintains a set of project plans and to-do lists, and coordinates the synchronization of both types of documents across changes made by project managers and individual contributors.

JAS allows push of information to the Java applets and allow them to push back with new information. Java applets are delivered via an HTTP server or through a Marimba Castanet Tuner. Once the applets are initialized, they connect directly to JAS for documents and authentication. JAS supports a flexible, remote procedure call-style protocol for client-server communication over TCP/IP. The RPC-style messages are constructed as Java objects, which are then communicated to the server. Communication between applets and JAS is accomplished by passing Java objects back and forth via a method called serialization. With serialization, an object graph is coded into a serialized stream of data, preserving all interobject references including circular references[7].

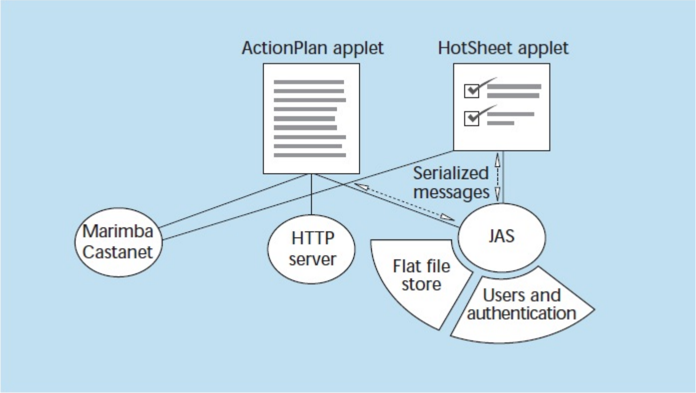


Figure 6: ActionPlan Architecture

G. Subversion

Subversion [11] (SVN) is a free/open source version control system (VCS) founded in 2000 by CollabNet Inc.. Subversion manages files and directories, and the changes made to them, over time [21]. This allows the user to recover older versions of data or examine the history of how the data changed.

Subversion can operate across networks, which allows it to be used by people on different computers. At some level, the ability for various people to modify and manage the same set of data from their respective locations fosters collaboration. Progress can occur more quickly without a single intermediary through which all modifications must occur.

Subversion is an appropriate tool for archiving old versions of files and directories, possibly resurrecting them, or examining logs of how those data changed over time. It can also be used to collaborate with people on documents and keep track of who made which changes. This is why Subversion is so often used in software development environments. Working on a development team is an inherently social activity, and Subversion makes it easy to collaborate with other programmers. However, despite its beneficial use, there is also a cost in using Subversion: administrative overhead. There is a need to manage a data repository to store the

[11] http://subversion.apache.org

information and all its history. Moreover, when working with the data on a daily basis, the user won't be able to copy, move, rename, or delete files the way it is usually done. Instead, all of those things is done through Subversion.

Figure 7 shows the architecture of subversion. On one end is a Subversion repository that holds all of the versioned data. On the other end is Subversion client program, which manages local reflections of portions of that versioned data. Between these extremes are multiple routes through a Repository Access (RA) layer, some of which go across computer networks and through network servers which then access the repository, others of which bypass the network altogether and access the repository directly.

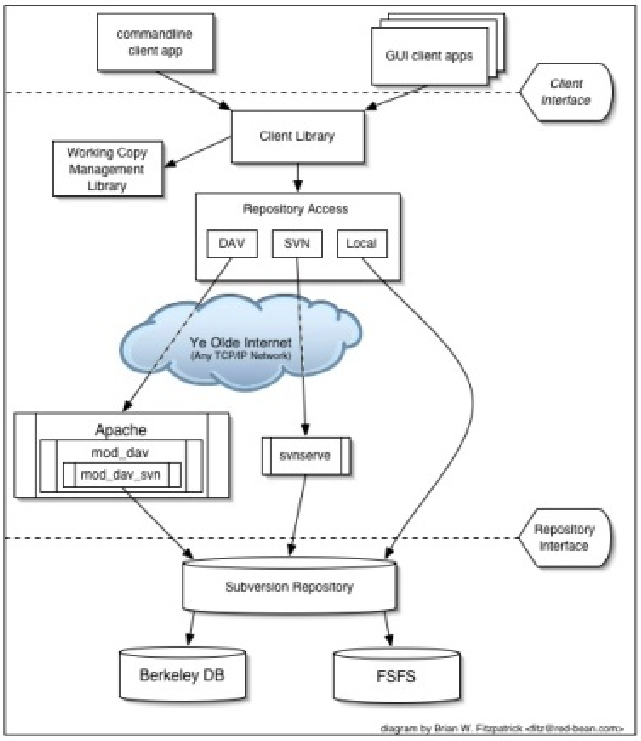


Figure 7: Subversion Architecture

IV. Design and Implementation

A. Context Diagram

The system has 6 access levels: system administrator, project administrator, project contributor, restricted user, registered user and non-registered user. The context diagram is shown in Figure 8.

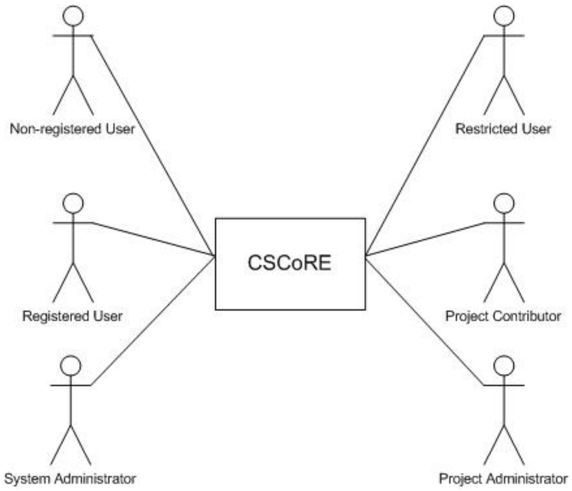


Figure 8: Context diagram of CSCoRE

B. Use Case Diagram

An non-registered user can browse the list of projects and request for an account in the system. Upon approval of the account request, he becomes a registered user. Aside from the capability to browse the list of projects, he can also request for membership in a project. He can also request for project creation. When the request is accepted, he automatically becomes the administrator for that project. The project administrator has the responsibility to manage the project and its members. The project members has two types: the restricted users and project contributors. Both have the access to the project but of different levels. Lastly, the system administrator is responsible for the management of the whole system. Figure 9 shows the top level use case diagram of CSCoRE.

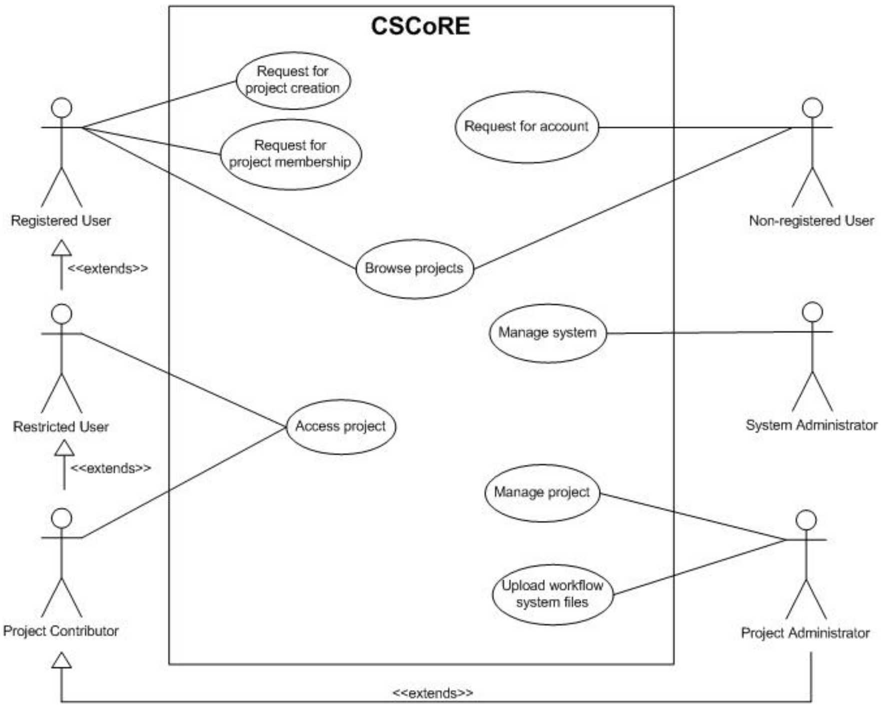


Figure 9: Top level use case diagram of CSCoRE

1. Request for Account

A non-registered user can request for a user account in the collaboratory. Figure 10 shows the request for account activity diagram.

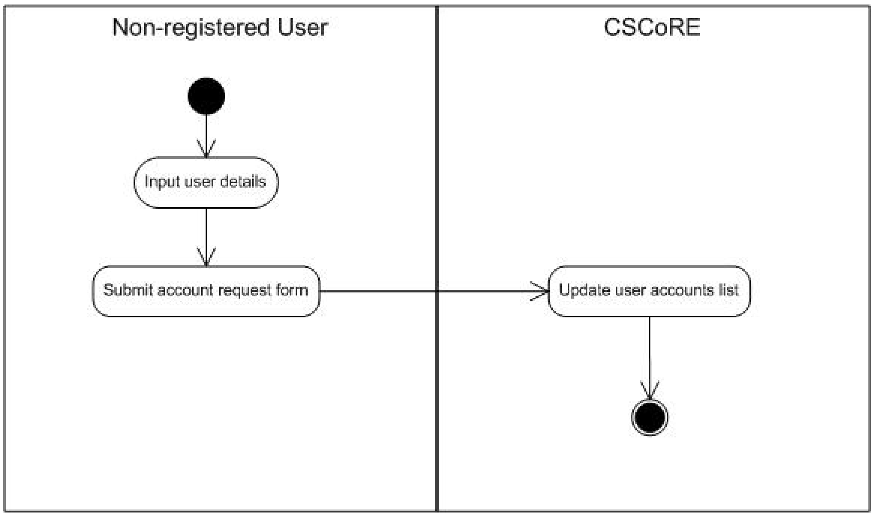


Figure 10: Request for account activity digram

2. Browse Projects

Both the registered and the non-registered users can search and view the list of projects. Upon selecting a project in the list, public information of the project is displayed. Figure 11 shows the browse projects activity diagram.

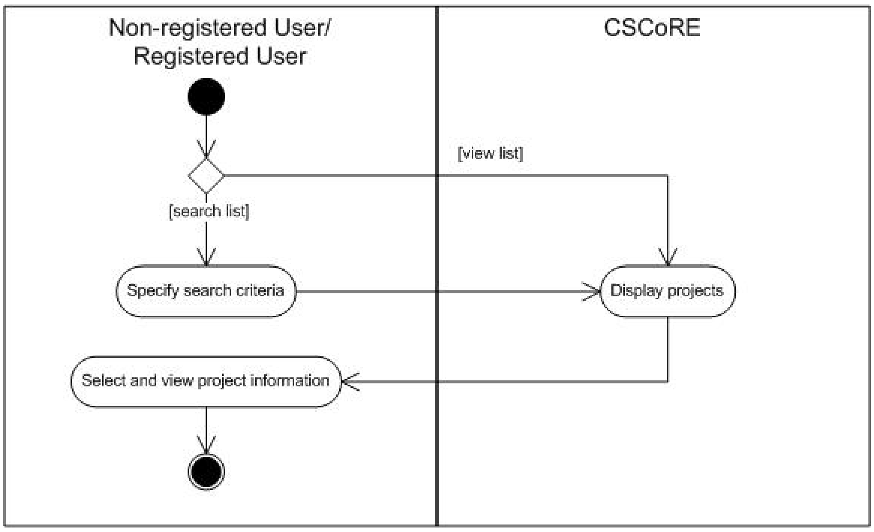


Figure 11: Browse projects activity digram

3. Request for Project Membership

A registered user can request for project membership in any of the projects listed in the system. Figure 12 shows the Request for project membership activity diagram.

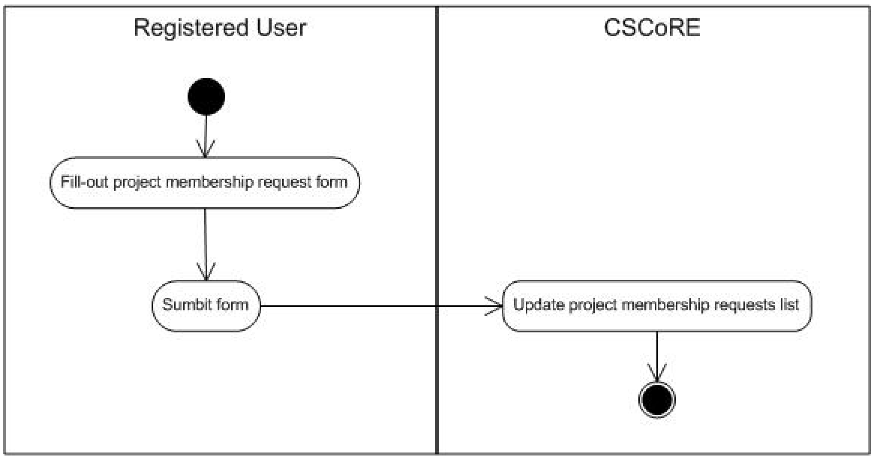

Registered User
CSCoRE
Fill-out project membership request form
Sumbit form
Update project membership requests list

Figure 12: Request for project membership activity digram

4. Request for Project Creation

A registered user can request for project creation. He can include a list of project members in the initial project information. Upon approval, the members of the new project is notified. Figure 13 shows the Request for project creation activity diagram.

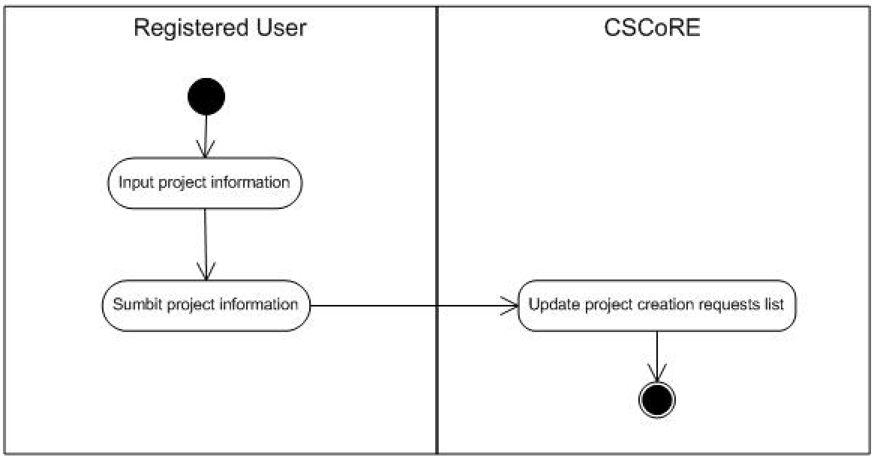

Registered User
CSCoRE
Input project information
Sumbit project information
Update project creation requests list

Figure 13: Request for project creation activity digram

5. Manage Project

The project administrator can edit the project information. He can also terminate the project. Furthermore, he also has the capability to manage the members of the project. Figure 14 shows the Manage project activity diagram.

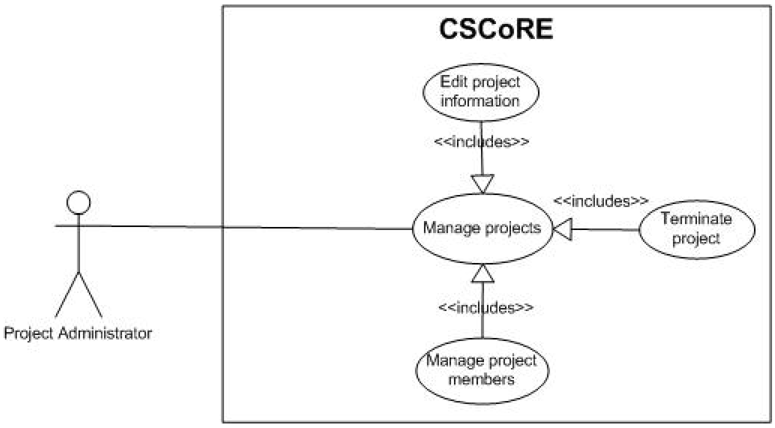


Figure 14: Manage project use case diagram

Edit Project Information The project administrator has the capability to edit the project details. Figure 15 shows the Edit project information activity diagram.

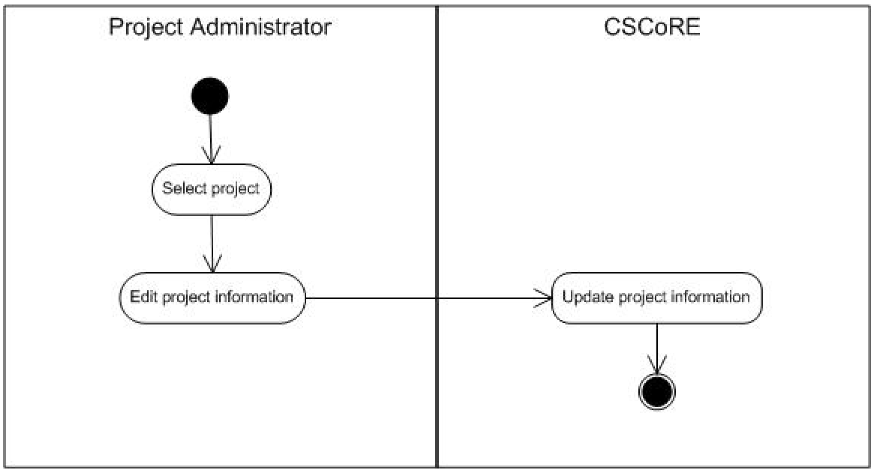


Figure 15: Edit project information activity digram

Terminate Project The project administrator has the capability to terminate the project. Figure 16 shows the Terminate project activity diagram.

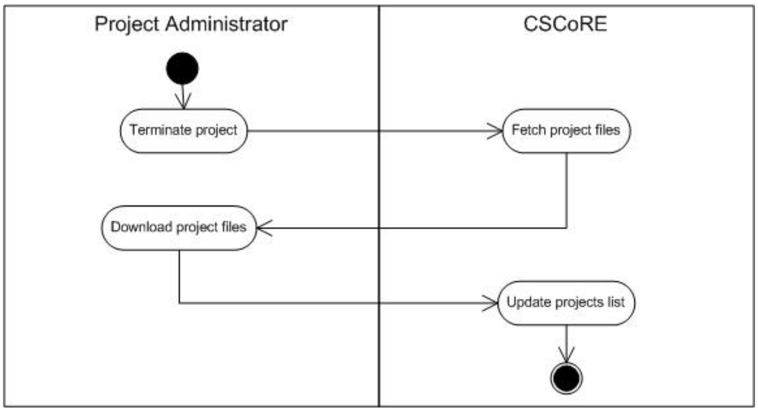


Figure 16: Terminate project activity digram

Manage Project Members The project administrator can approve project membership requests, add and delete members of the project. He can also edit the membership of each member. The project members and the project administrator can view the members of the project. Figure 17 shows the Manage project members use case diagram, Figure 18 shows the Add project member activity diagram, Figure 19 shows the Edit membership activity diagram, Figure 21 shows the View project members activity diagram, Figure 20 shows the Remove project member activity diagram and Figure 22 shows the Accept request for project membership activity diagram.

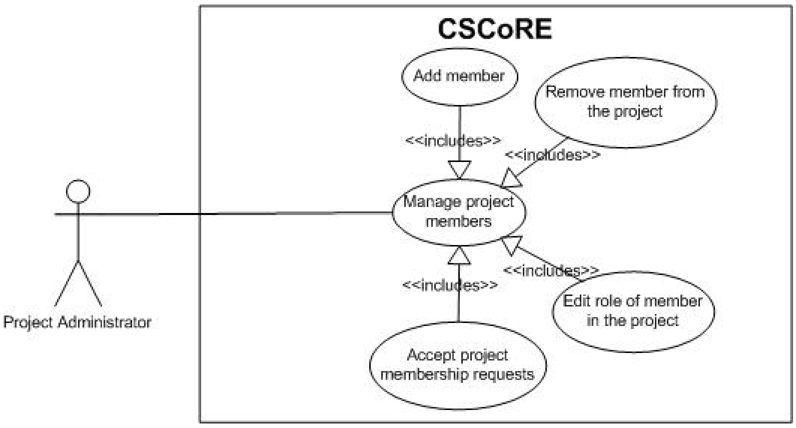


Figure 17: Manage project members use case diagram

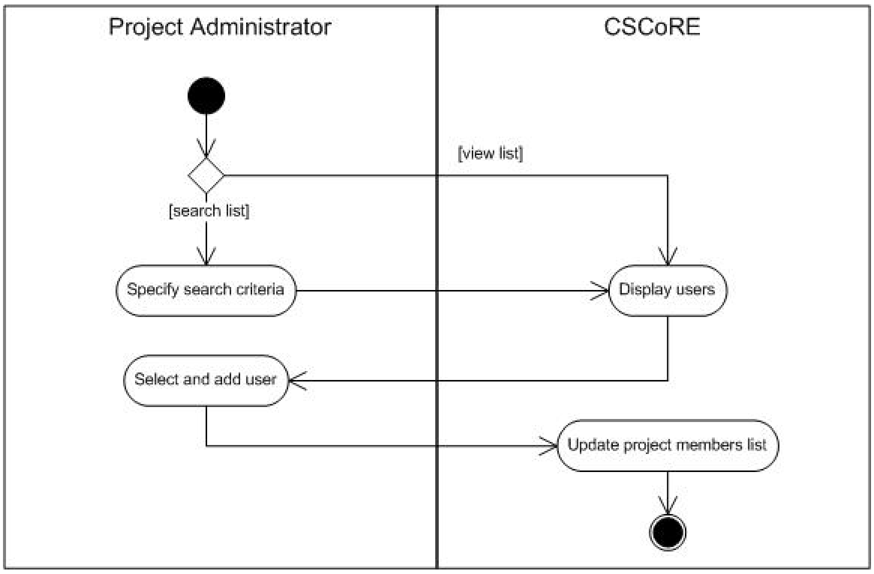


Figure 18: Add member activity digram

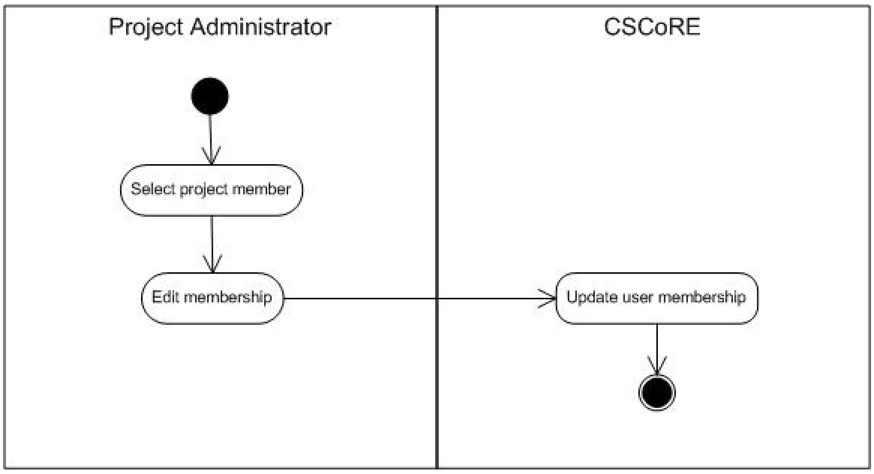


Figure 19: Edit membership activity digram

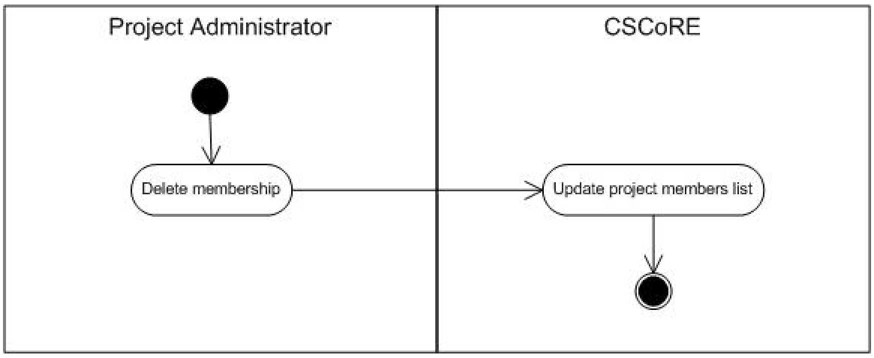


Figure 20: Remove project member activity digram

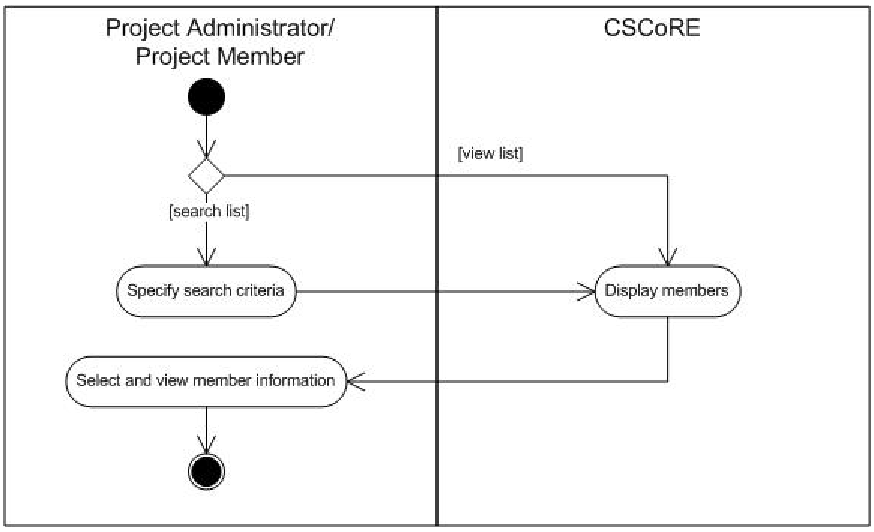


Figure 21: View member activity digram

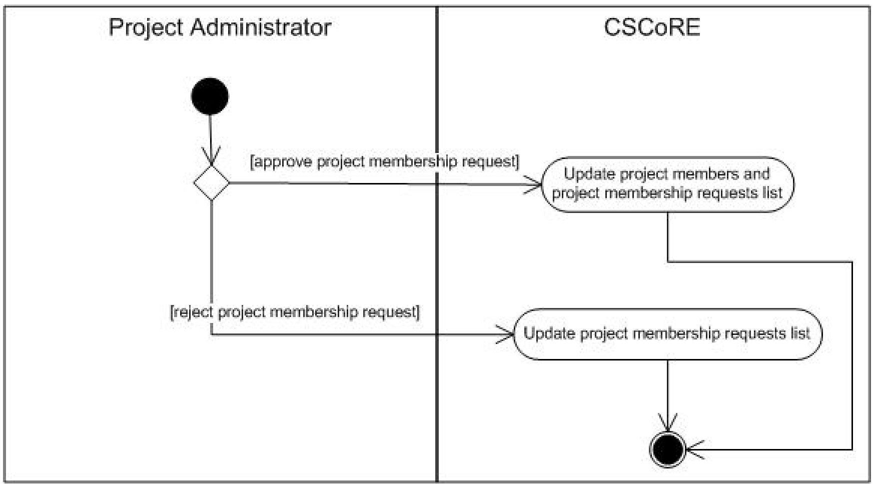

Project Administrator
CSCoRE
[approve project membership request]
Update project members and project membership requests list
[reject project membership request]
Update project membership requests list

Figure 22: Accept request for project membership activity digram

6. Access Project

The project contributor has the privilege to manage miscellaneous files, tools, and references. He can also participate in forums and access files in the subversion repository. The restricted user on the other hand can only download the materials in the project. Figure 23 shows the Access project use case diagram.

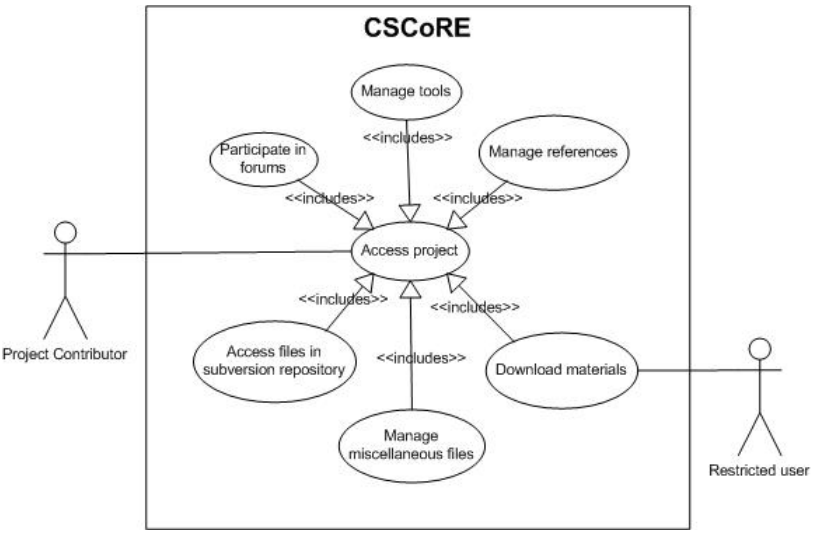


Figure 23: Access project use case diagram

Manage Miscellaneous Files The project contributor can add, edit, and delete any miscellaneous file. Figure 24 shows the Manage miscellaneous files use case diagram, Figure 25 shows the Add miscellaneous file activity diagram, Figure 26 shows the Edit miscellaneous file activity diagram, and Figure 27 shows the Delete miscellaneous file activity diagram.

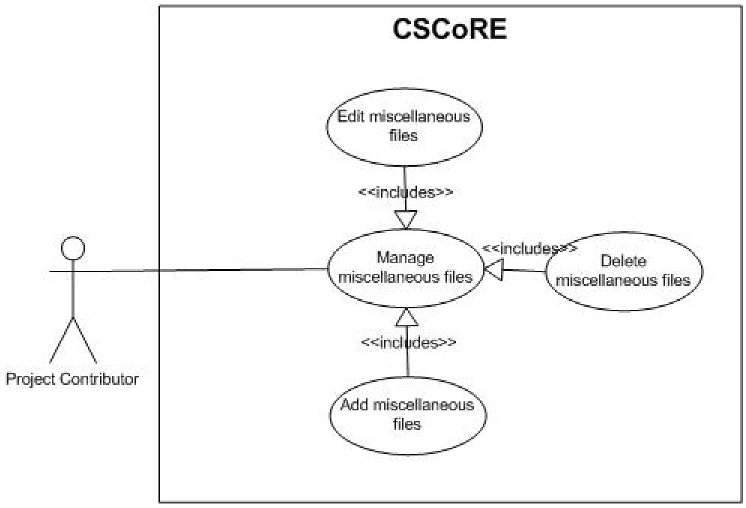


Figure 24: Manage miscellaneous files use case diagram

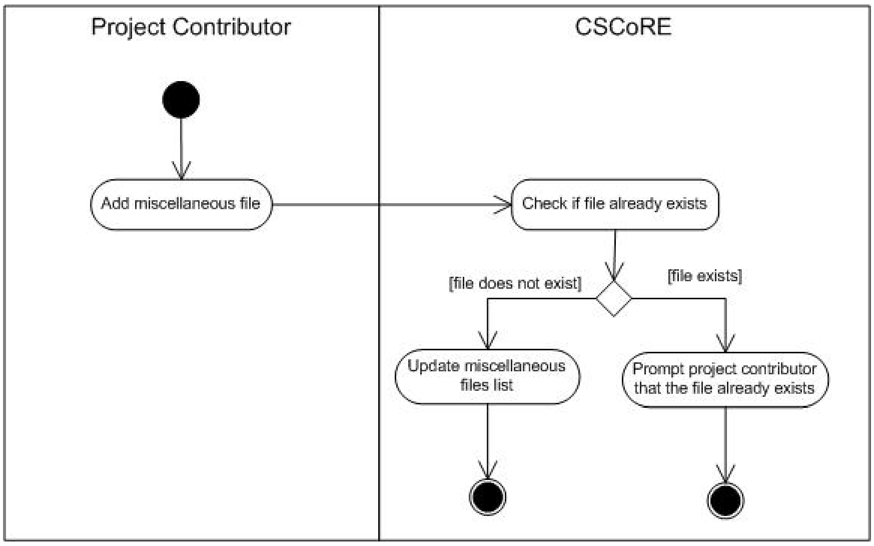

Project Contributor
CSCoRE
Add miscellaneous file
Check if file already exists
[file does not exist]
[file exists]
Update miscellaneous files list
Prompt project contributor that the file already exists

Figure 25: Add miscellaneous file activity digram

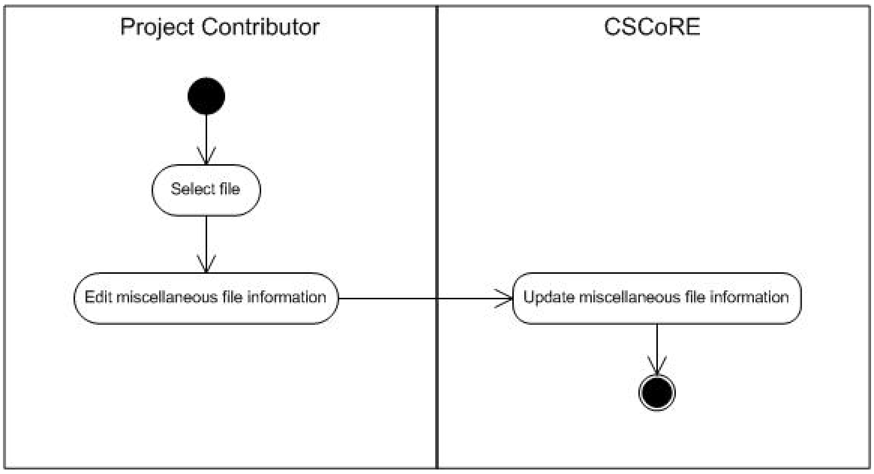

Project Contributor
CSCoRE
Select file
Edit miscellaneous file information
Update miscellaneous file information

Figure 26: Edit miscellaneous file activity digram

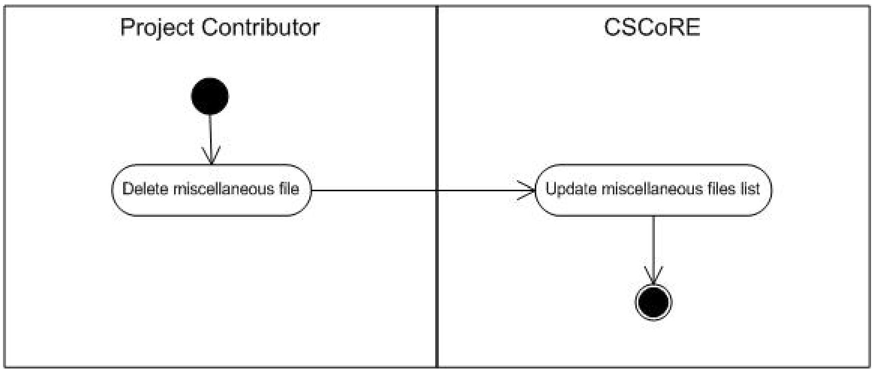


Figure 27: Delete miscellaneous file activity digram

Manage Tools A project contributor can add, edit and delete tools. Figure 28 shows the Manage tools use case diagram, Figure 29 shows the Add tool activity diagram, Figure 30 shows the Edit tool activity diagram, and Figure 31 shows the Delete tool activity diagram.

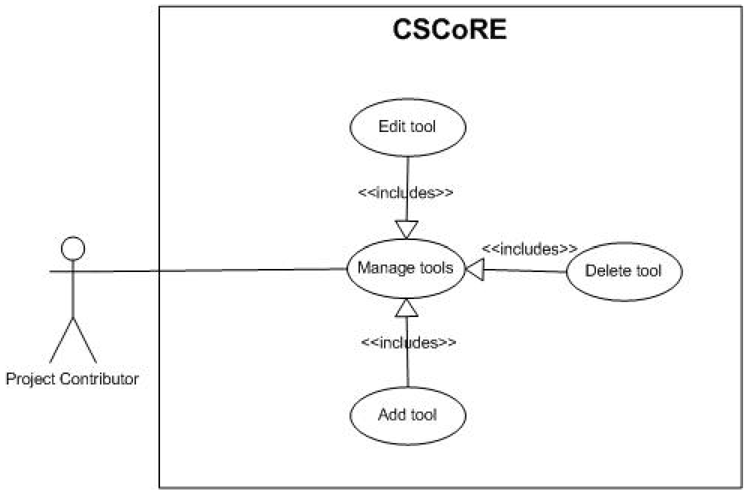


Figure 28: Manage tools use case diagram

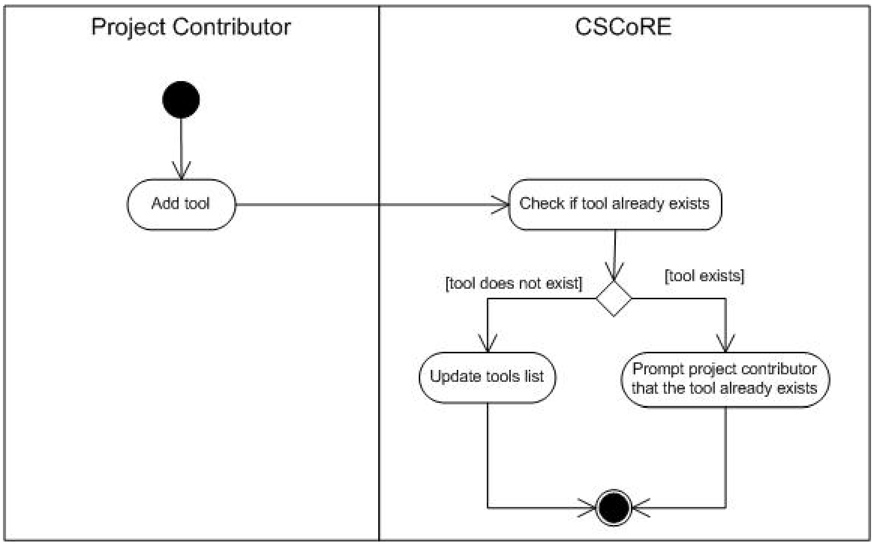


Figure 29: Add tool activity digram

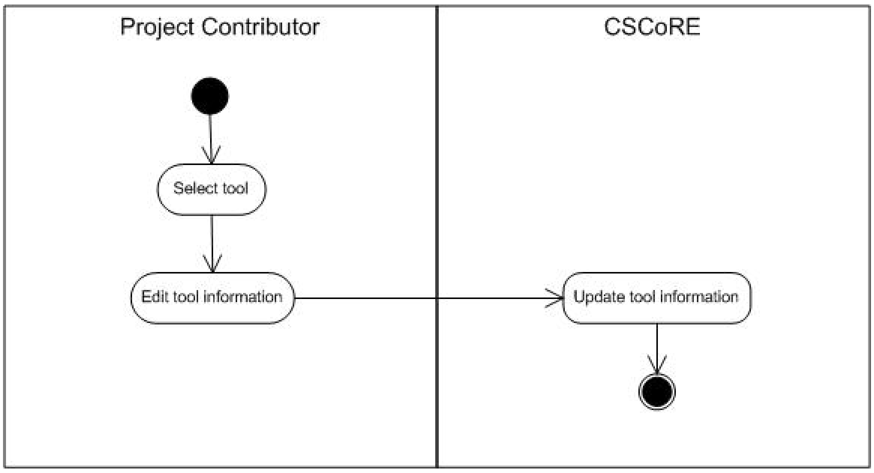


Figure 30: Edit tool activity digram

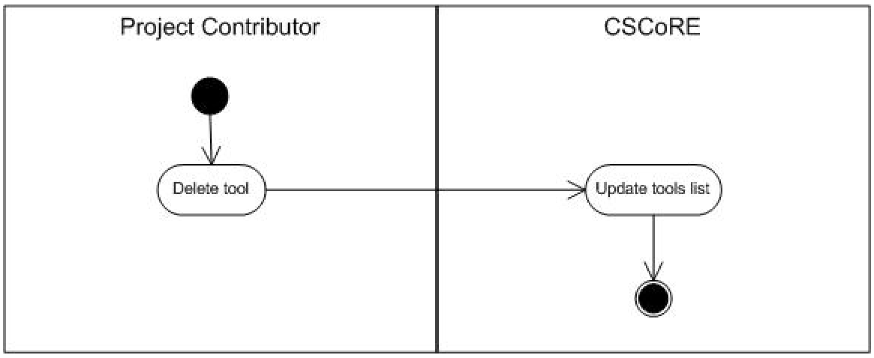


Figure 31: Delete tool activity digram

Manage References A project contributor can add, edit and delete references. He can also make a comment to an uploaded reference. Figure 32 shows the Manage references use case diagram, Figure 33 shows the Add reference activity diagram, Figure 34 shows the Edit reference activity diagram, Figure 35 shows the Delete reference activity diagram and Figure 36 shows the Comment to an uploaded reference activity diagram.

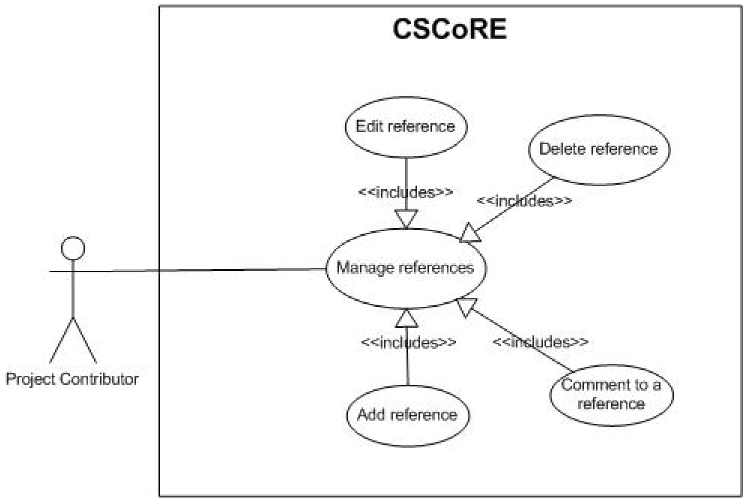


Figure 32: Manage references use case diagram

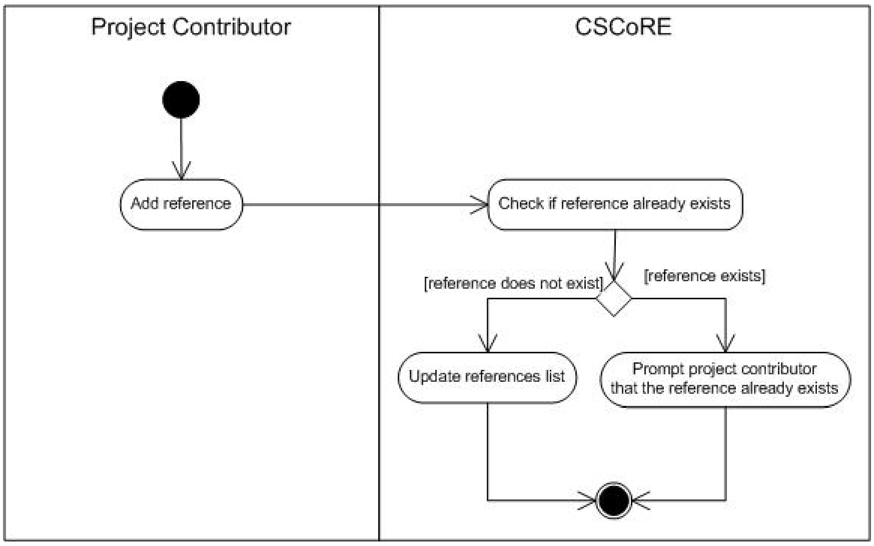

Project Contributor
CSCoRE
Add reference
Check if reference already exists
[reference does not exist]
[reference exists]
Update references list
Prompt project contributor that the reference already exists

Figure 33: Add reference activity digram

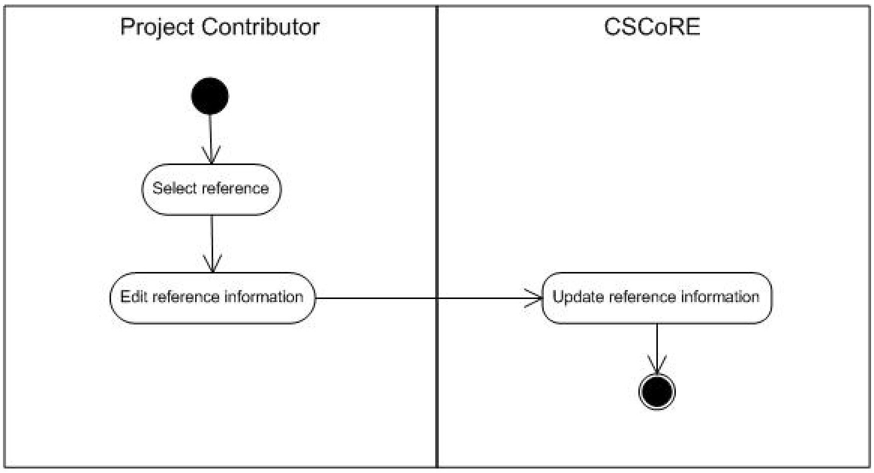


Figure 34: Edit reference activity digram

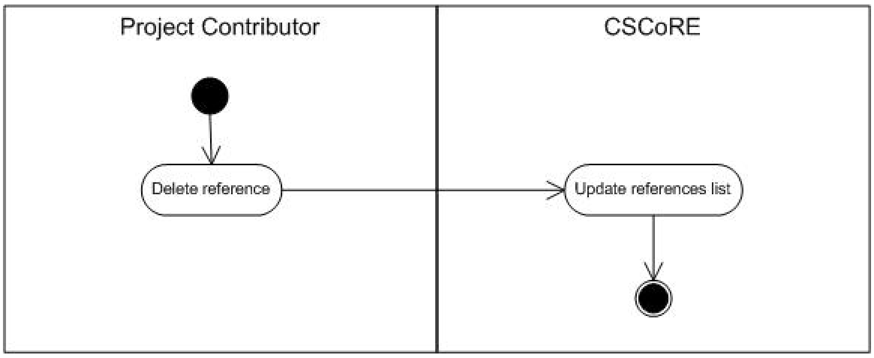


Figure 35: Delete reference activity digram

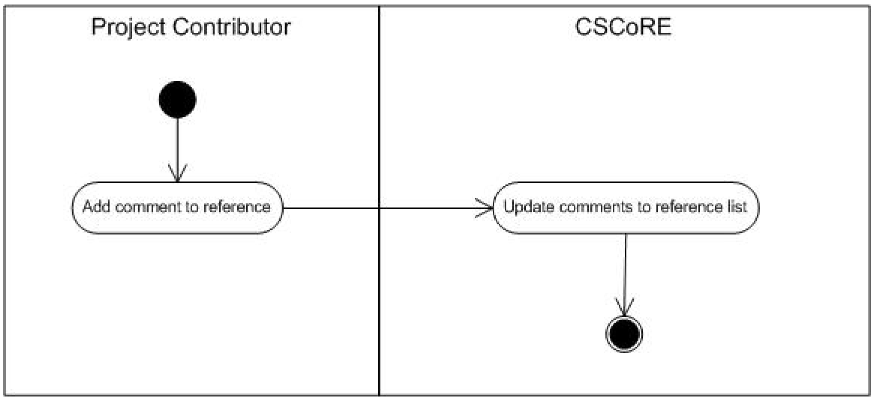


Figure 36: Comment to an uploaded reference activity diagram

Participate in the Forum A project contributor can view threads, post new threads and post replies in the forum. Figure 37 shows the Participate in the forum use case diagram, Figure 38 shows the View thread activity diagram, Figure 39 shows the Post new thread activity diagram, and Figure 40 shows the post reply activity diagram.

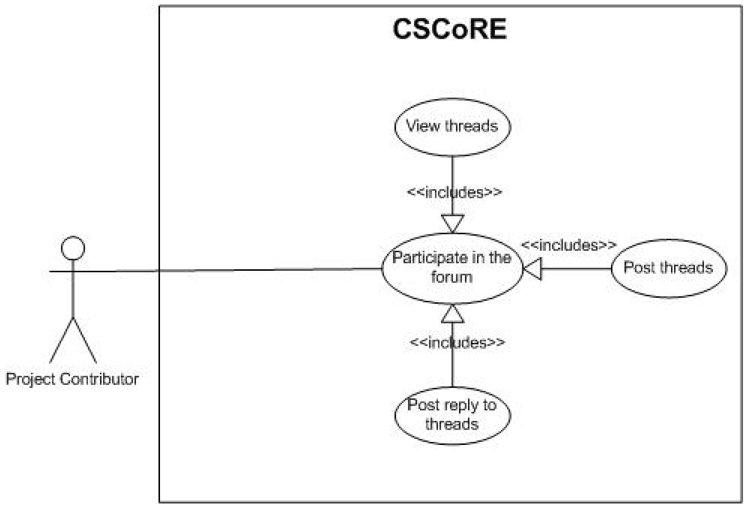


Figure 37: Participate in the forum use case diagram

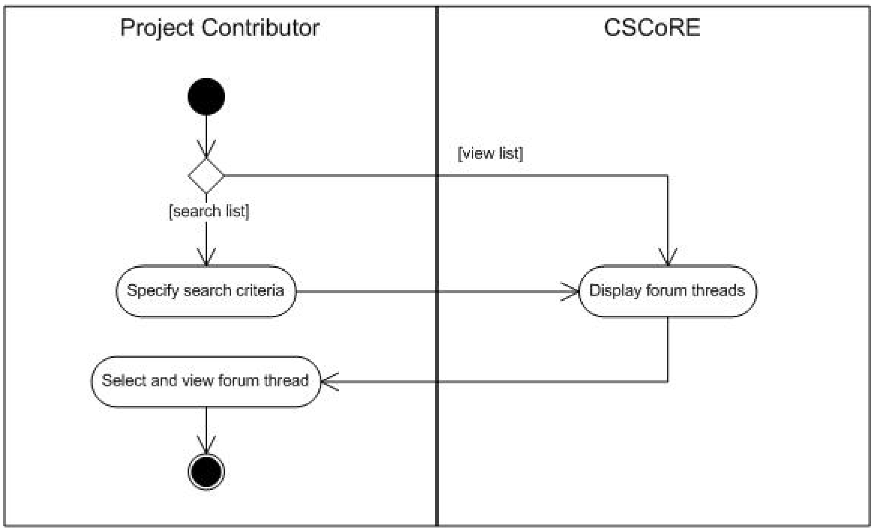


Figure 38: View thread activity digram

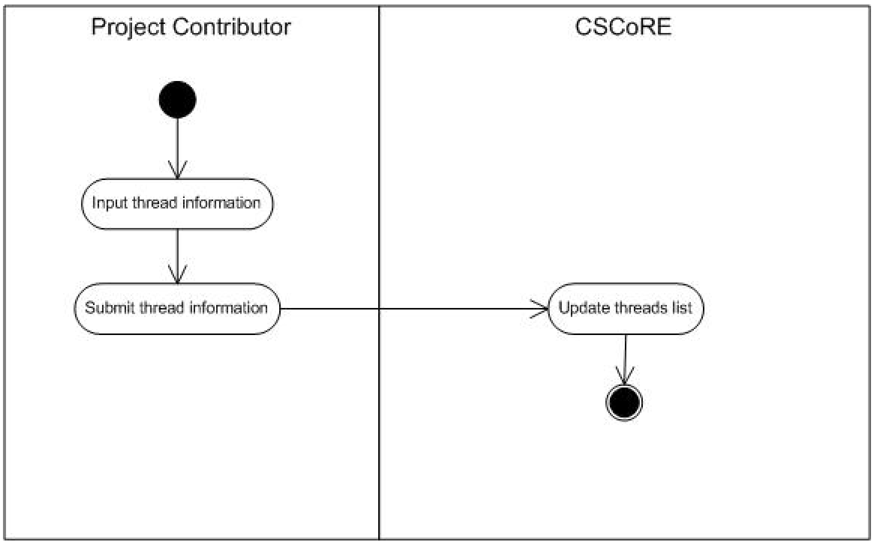


Figure 39: Post new thread activity digram

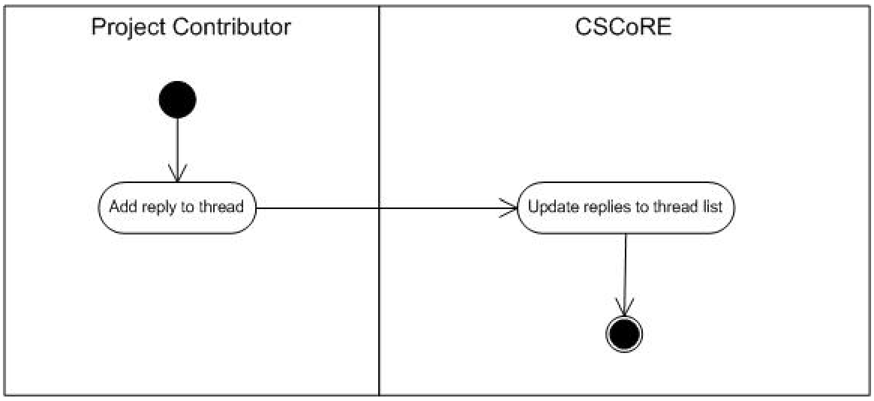


Figure 40: Post reply activity digram

Access Files in Subversion Repository A project contributor can upload a file, check-out a file and commit changes in the files in the subversion repository. Figure 41 shows the Access files in subversion repository use case diagram, Figure 42 shows the Upload file activity diagram, Figure 43 shows the Check-out file activity diagram, and Figure 44 shows the Commit changes to file activity diagram.

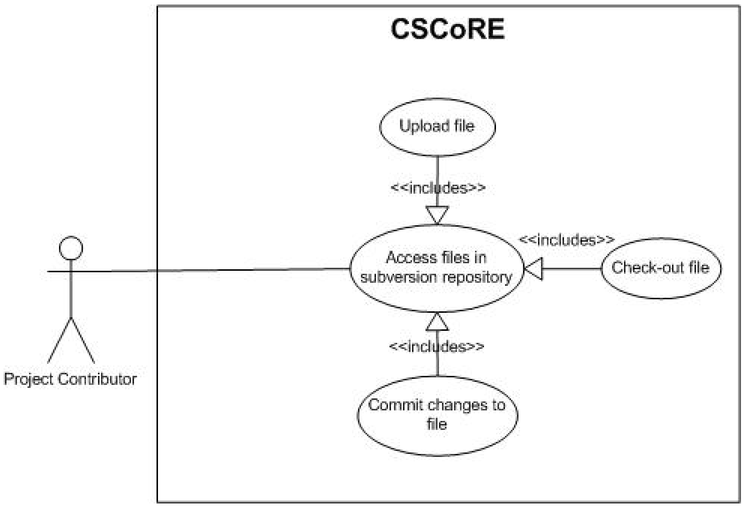


Figure 41: Access files in subversion repository use case diagram

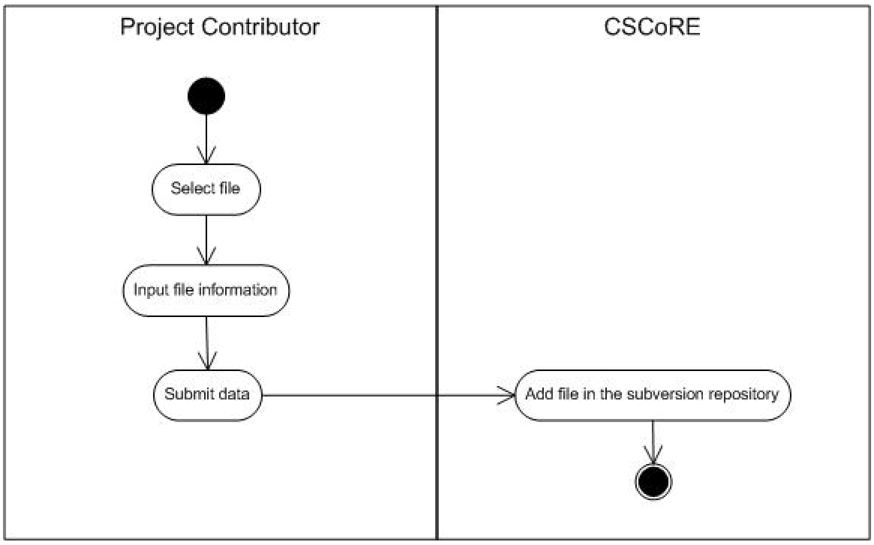


Figure 42: Upload file activity diagram

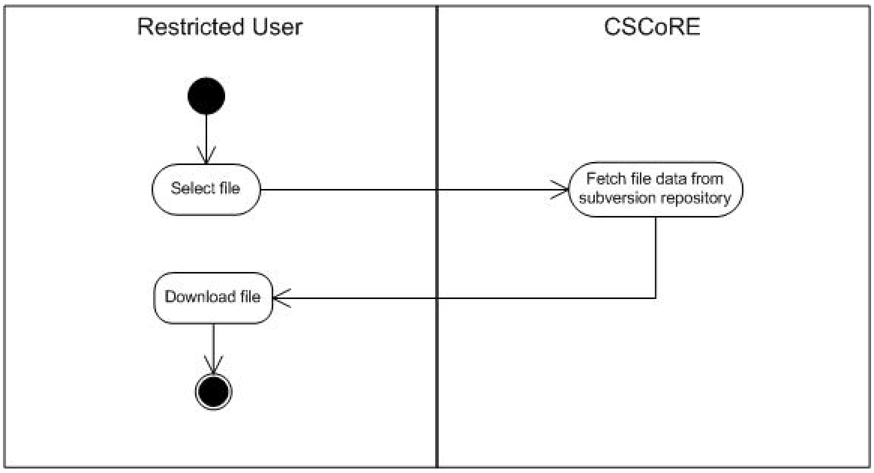


Figure 43: Check-out file activity diagram

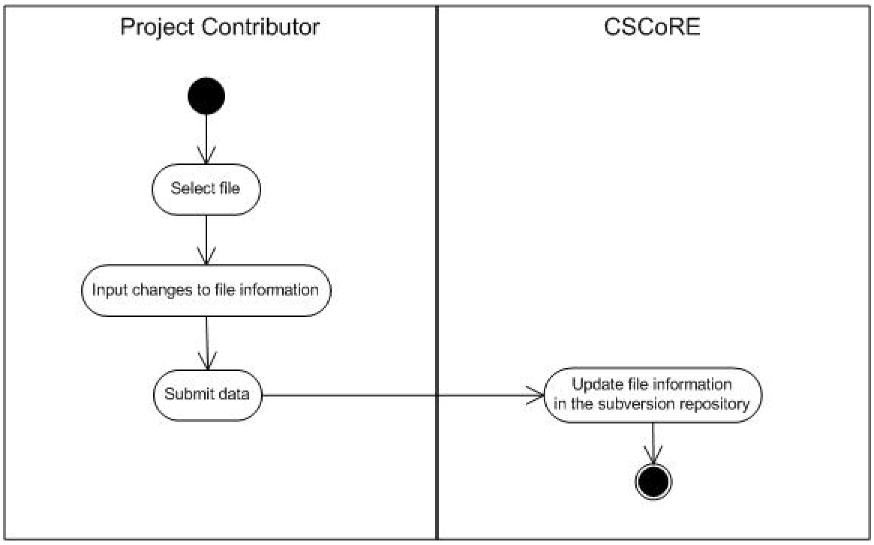

Project Contributor
CSCoRE
Select file
Input changes to file information
Submit data
Update file information in the subversion repository

Figure 44: Commit changes to file activity diagram

Download Materials The restricted user has the capability to view and download miscellaneous files, tools and references. Figure 45 shows the download materials use case diagram, Figure 46 and Figure 47 shows the View and Download miscellaneous file activity diagram, Figure 48 and Figure 49 shows the View and Download tool activity diagram, and Figure 50 and Figure 51 shows the View and Download reference activity diagram.

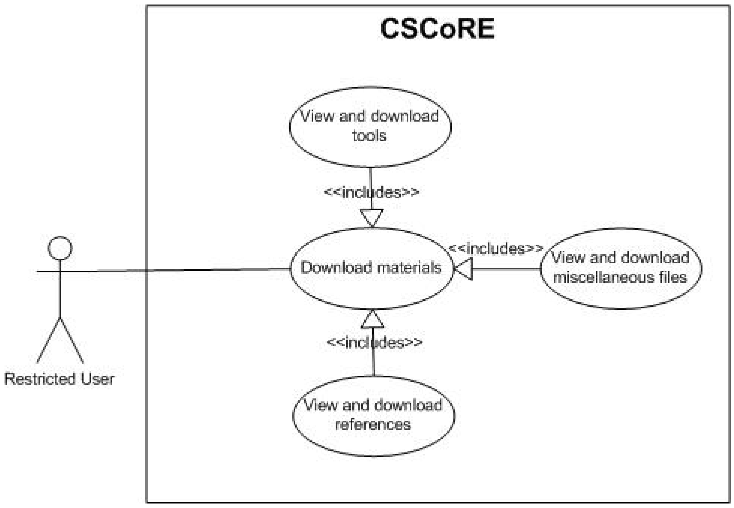


Figure 45: Download materials use case diagram

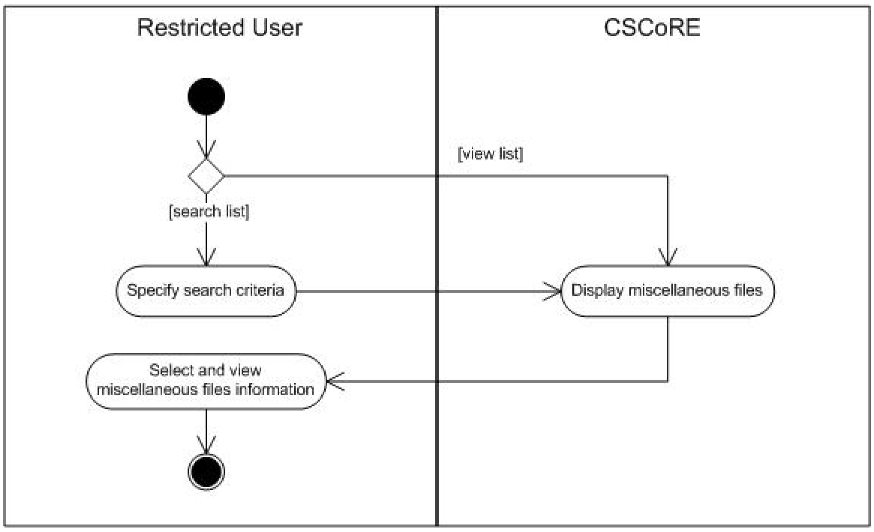

Restricted User
CSCoRE
[view list]
[search list]
Specify search criteria
Display miscellaneous files
Select and view miscellaneous files information

Figure 46: View miscellaneous file activity digram

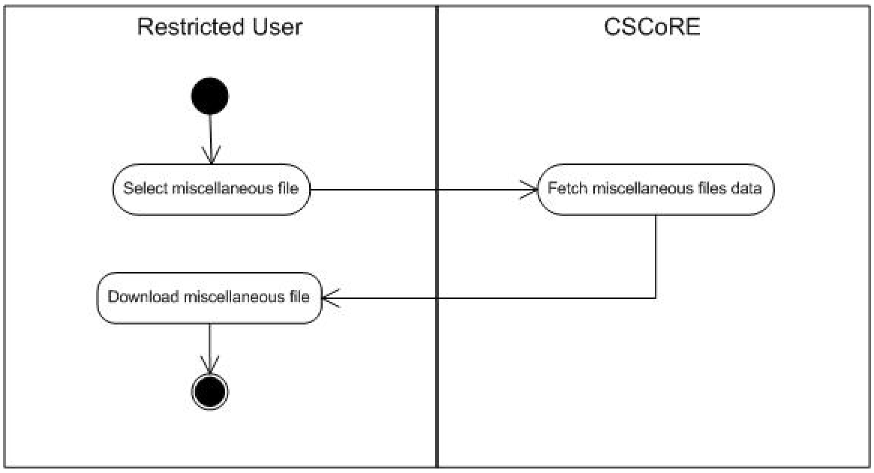

Restricted User
CSCoRE
Select miscellaneous file
Fetch miscellaneous files data
Download miscellaneous file

Figure 47: Download miscellaneous file activity digram

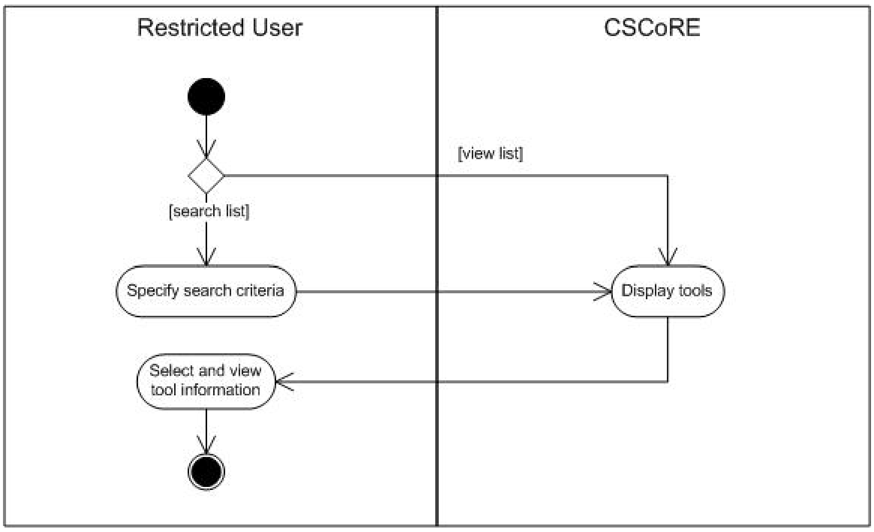


Figure 48: View tool activity digram

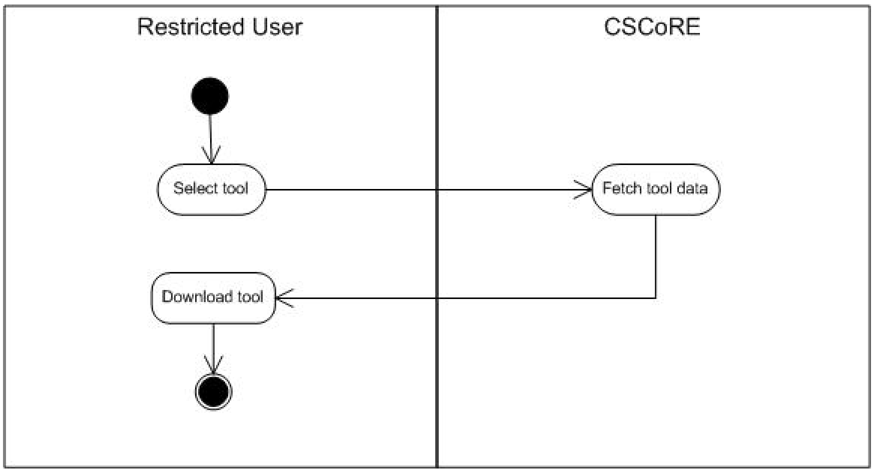


Figure 49: Download tool activity digram

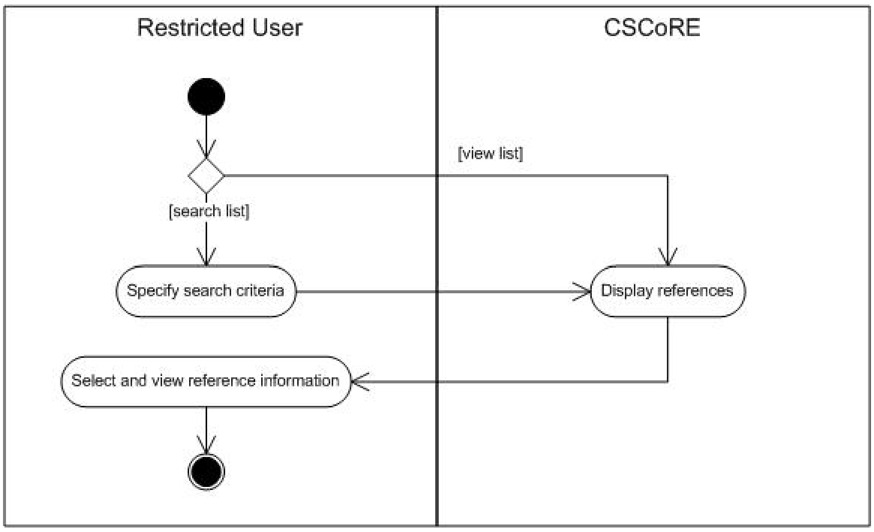


Figure 50: View reference activity digram

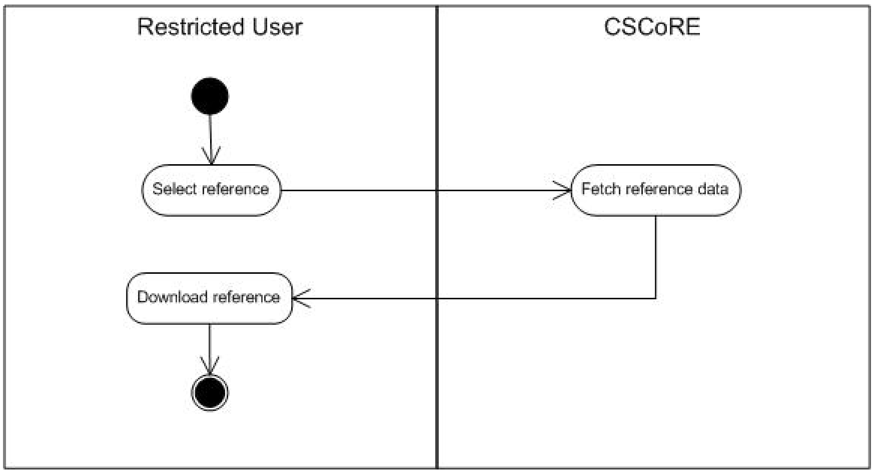


Figure 51: Download reference activity digram

7. Manage System

The system administrator has the capability to manage the system which includes managing the projects list and users. Figure 52 shows the Manage system use case diagram.

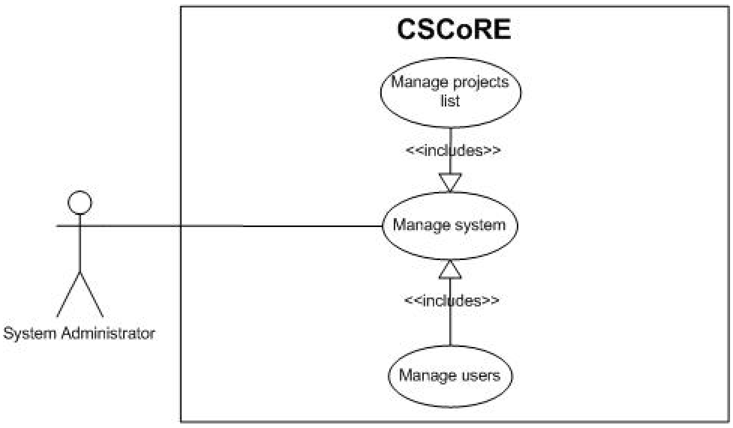


Figure 52: Manage system use case diagram

Manage Projects List Only the system administrator can add and delete projects and approve project creation requests in the system. Figure 53 shows the Manage projects use case diagram, Figure 54 shows the Add project activity diagram, Figure 55 shows the Delete project activity diagram and Figure 56 shows the Approve project creation request activity diagram.

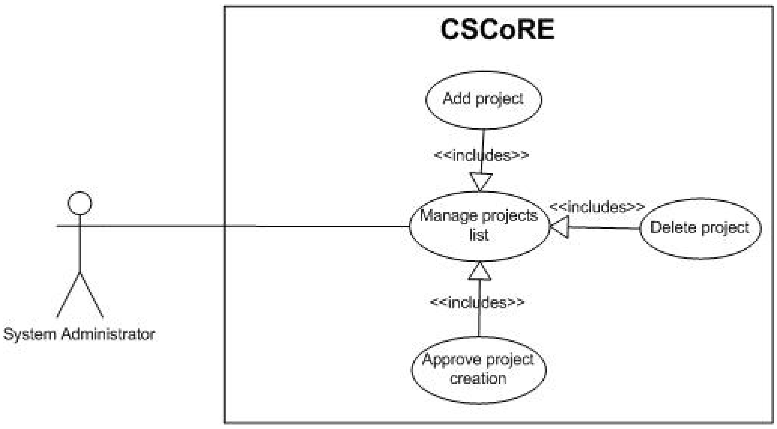


Figure 53: Manage projects use case diagram

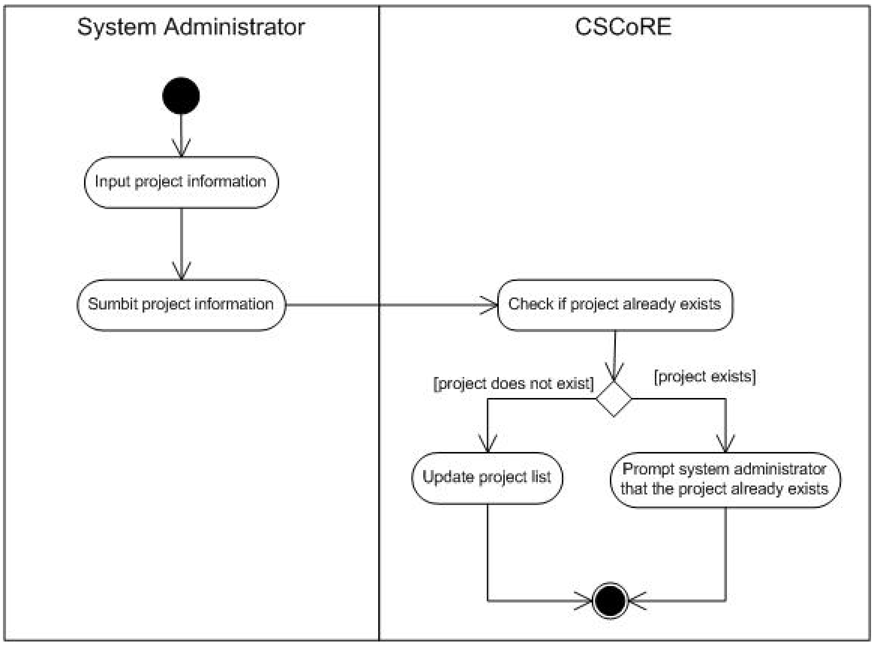

System Administrator
CSCoRE
Input project information
Sumbit project information
Check if project already exists
[project does not exist]
[project exists]
Update project list
Prompt system administrator that the project already exists

Figure 54: Add project activity digram

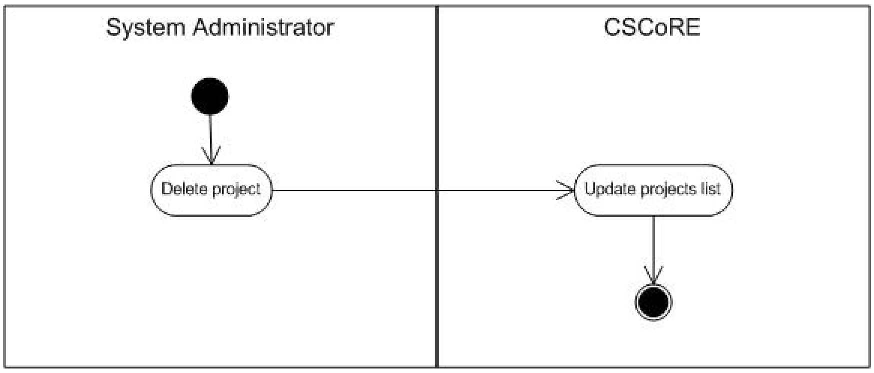


Figure 55: Delete project activity digram

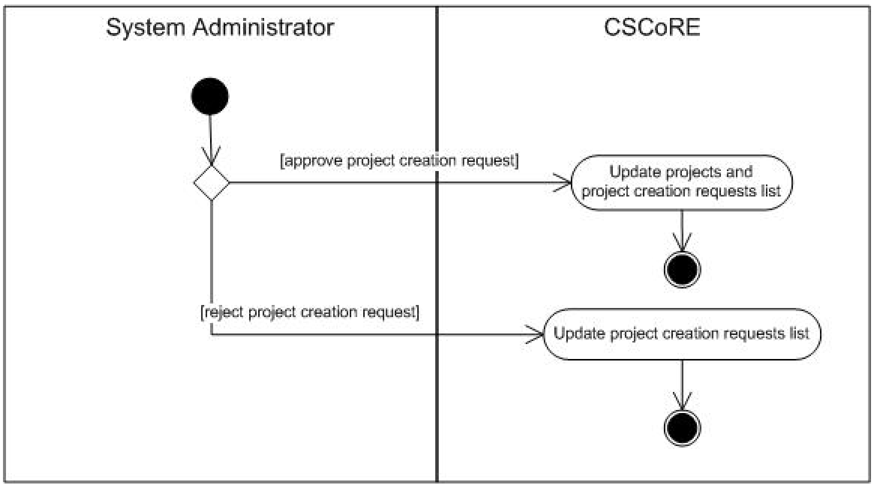

System Administrator
CSCoRE
[approve project creation request]
Update projects and project creation requests list
[reject project creation request]
Update project creation requests list

Figure 56: Approve project creation request activity digram

Manage Users The system administrator can add, edit and view user accounts in the system. He can also accpect account creation requests. Moreover, he can delete user accounts with the exception of the accounts of project administrators of active projects. Figure 57 shows the Manage users use case diagram, Figure 58 shows the Add user accounr activity diagram, Figure 59 shows the Edit user account activity diagram, Figure 60 shows the Delete user account activity diagram, Figure 61 shows the View user account activity diagram and Figure 62 shows the Accept user account request activity diagram.

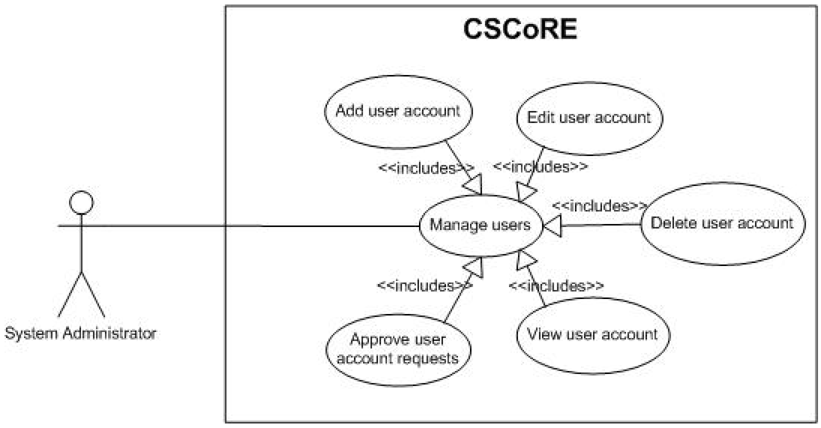


Figure 57: Manage users use case diagram

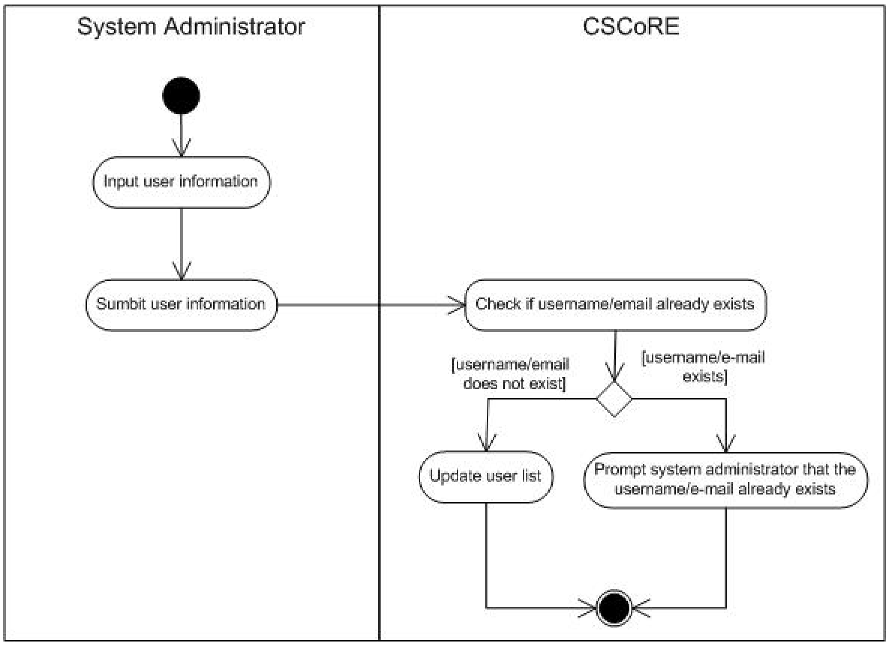

System Administrator
CSCoRE
Input user information
Sumbit user information
Check if username/email already exists
[username/email does not exist]
[username/e-mail exists]
Update user list
Prompt system administrator that the username/e-mail already exists

Figure 58: Add user account activity digram

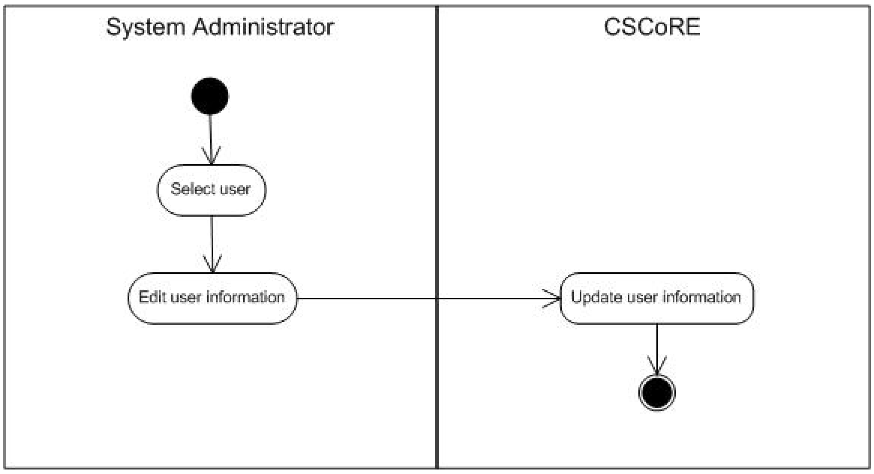


Figure 59: Edit user account activity digram

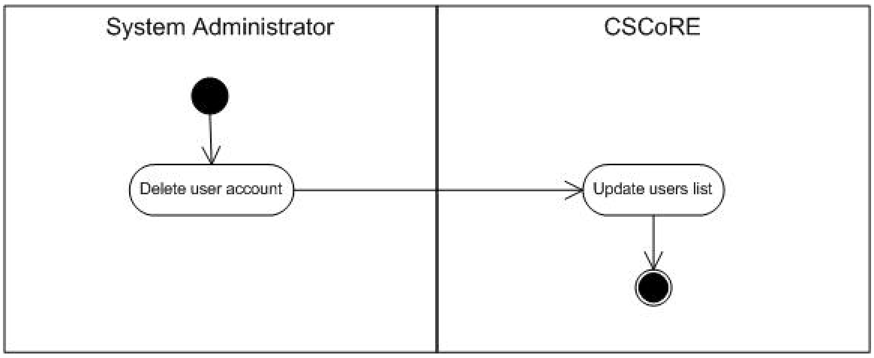


Figure 60: Delete user account activity digram

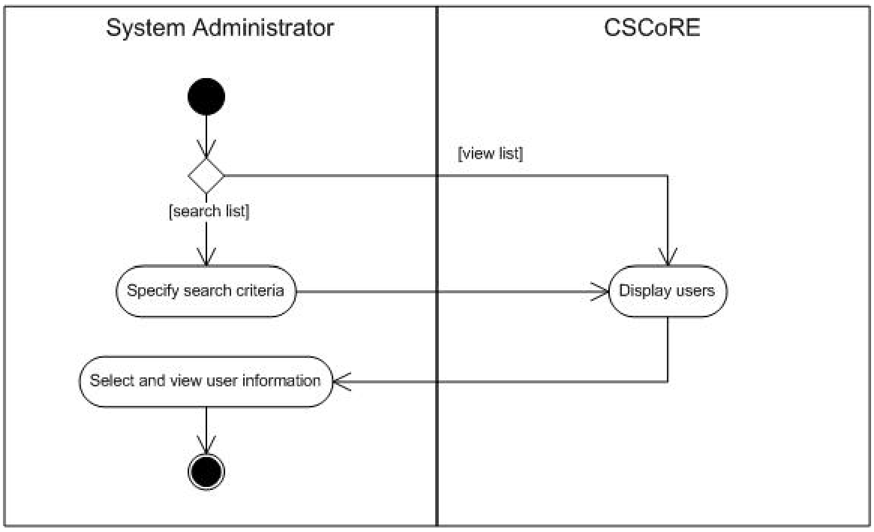


Figure 61: View user account activity digram

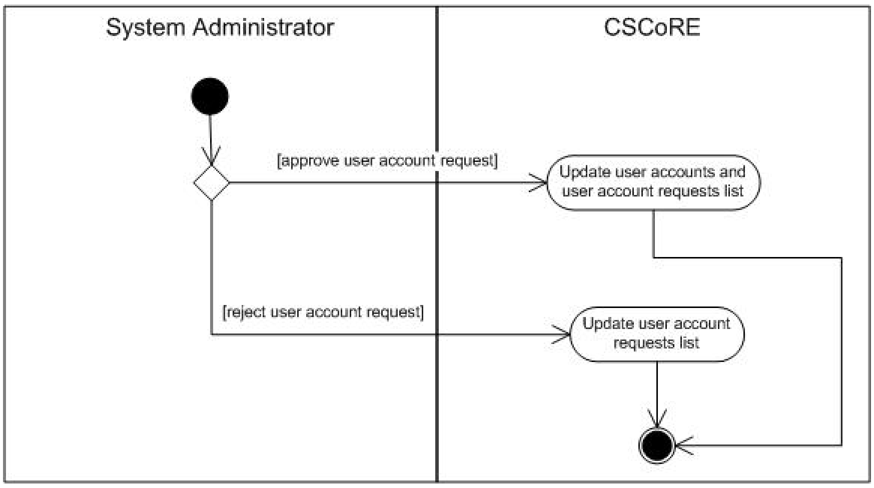

System Administrator
CSCoRE
[approve user account request]
Update user accounts and user account requests list
[reject user account request]
Update user account requests list

Figure 62: Approve user account request activity digram

C. Entity Relationship Diagram

Figure 63 shows the Entity Relationship Diagram of CSCoRE.

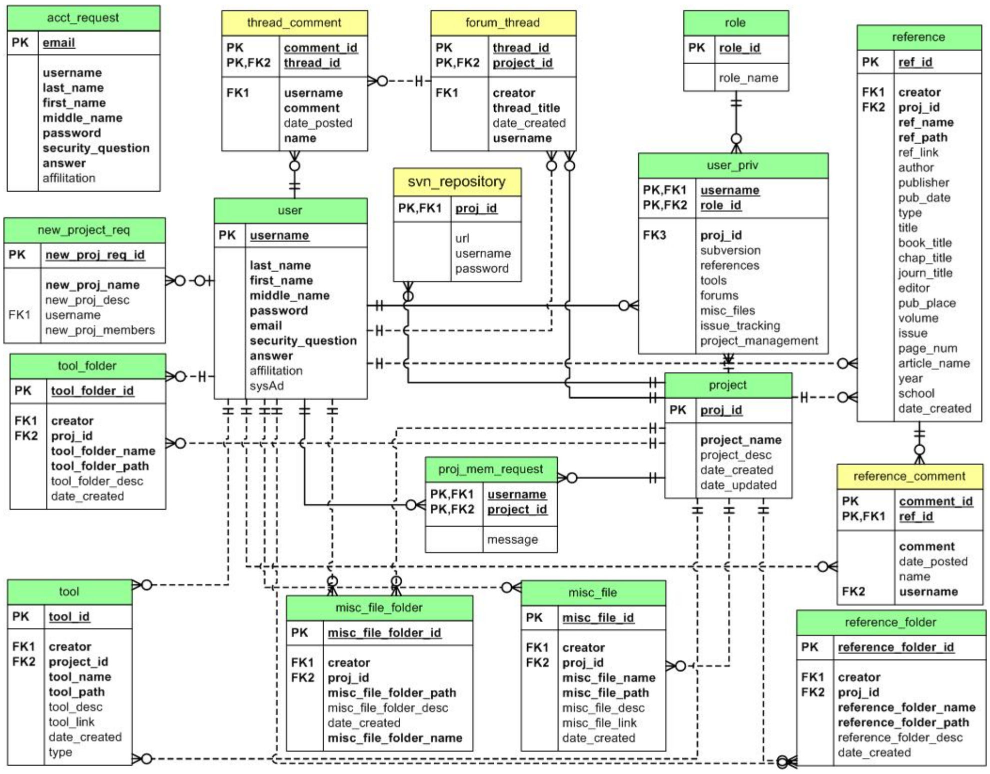


Figure 63: Entity Relationship Diagram of CSCoRE

D. Data Dictionary

Listed below are the tables in CSCoRE and their respective data fields.

user- contains the account details and basic information of the user

Field	Type	Description
username	VARCHAR(255)	Unique identifier of user
last_name	VARCHAR(50)	Last name of user
first_name	VARCHAR(50)	First name of user
middle_name	VARCHAR(50)	Middle name of user
password	VARCHAR(100)	Password of user
email	VARCHAR(50)	E-mail address of user
security_question	VARCHAR(100)	Security question
answer	VARCHAR(100)	Answer to the security question
affiliation	VARCHAR(50)	Institution/organization where the user belongs
sysAd	set('Yes','No')	Indicates the user is a system administrator

Table 2: user table

user_priv- contains the roles of a user in the system

Field	Type	Description
username	VARCHAR(255)	Unique identifier of user
proj_id	INTEGER	Project where the role of the user is applied
role_id	set('0','1')	Main role in the project
subversion	set('1','2','3')	Role the subversion module

references	set('1','2','3')	Role in the references module
tools	set('1','2','3')	Role in the tools module
forums	set('1','2','3')	Role in the forums module
misc_files	set('1','2','3')	Role in the miscellaneous files module
issue_tracking	set('1','2','3')	Role in the issue tracking module
proj_management	set('1','2','3')	Role in the project management module

Table 3: user_priv table

role- list of the roles in the system

Field	Type	Description
role_id	INTEGER	Code assigned to a role
role_name	VARCHAR(20)	Name of role

Table 4: role table

new_proj_request- list of project creation requests

Field	Type	Description
new_proj_req_id	INTEGER	Unique identifier of a project creation request
new_proj_name	VARCHAR(200)	Name of project
new_proj_desc	TEXT	Description of project
username	VARCHAR(255)	Username of the requestor
new_proj_members	TEXT	List of initial project members

Table 5: new_proj_request table

proj_mem_request- list of project membership requests

Field	Type	Description
username	VARCHAR(255)	Username of the requestor
proj_id	INTEGER	Project where membership is rquested
message	VARCHAR(200)	Message containing reasons to approve membership requests

Table 6: proj_mem_request table

acct_request- list of user account requests

Field	Type	Description
email	VARCHAR(50)	E-mail address of requestor
username	VARCHAR(255)	Unique identifier of user
last_name	VARCHAR(50)	Last name of user
first_name	VARCHAR(50)	First name of user
middle_name	VARCHAR(50)	Middle name of user
password	VARCHAR(100)	Password of user
security_question	VARCHAR(100)	Security question
answer	VARCHAR(50)	Answer to the security question
affiliation	VARCHAR(50)	Institution/organization where the user belongs

Table 7: acct_request table

project- list of projects in the system

Field	Type	Description
project_id	INTEGER	Unique identifier of the project
proj_name	VARCHAR(200)	Name of project
proj_desc	TEXT	Description of project
date_created	DATETIME	Date when project was created
date_updated	DATETIME	Date when project was updated

Table 8: project table

reference_folder- list of references folders of the projects in the system

Field	Type	Description
ref_folder_id	INTEGER	Unique identifier of the references folder
ref_folder_name	VARCHAR(200)	Name of references folder
ref_folder_desc	TEXT	Description of references folder
ref_folder_path	VARCHAR(200)	Path where references folder can be found
creator	VARCHAR(255)	User who created the references folder
proj_id	INTEGER	Project where the references folder belongs

Table 9: reference_folder table

reference- list of references of the projects in the system

Field	Type	Description
ref_id	INTEGER	Unique identifier of the reference
ref_name	VARCHAR(200)	Name of reference
ref_link	VARCHAR(200)	Link of other information about the reference
ref_path	VARCHAR(200)	Path where reference can be found
creator	VARCHAR(255)	User who uploaded the reference
proj_id	INTEGER	Project where the reference belongs
type	INTEGER	Type of reference
tile	TEXT	Title of reference
book_title	TEXT	Title of book
chap_title	VARCHAR(200)	Title of chapter
journ_title	VARCHAR(200)	Title of journal
author	VARCHAR(200)	Author of reference
publisher	VARCHAR(200)	Publisher of reference
pub_date	DATE	Date when reference was published
pub_place	VARCHAR(200)	Place where reference was published
editor	VARCHAR(100)	Editor of reference
volume	VARCHAR(20)	Volume of journal
issue	VARCHAR(20)	Issue of journal
page_num	VARCHAR(20)	Page where reference was found
article_name	TEXT	Name of article
year	INTEGER	Year reference was published
school	VARCHAR(100)	School where thesis was submitted
date_created	DATETIME	Date when reference was added

Table 10: reference table

reference_comment- list of comments to the references

Field	Type	Description
comment_id	INTEGER	Unique identifier of the comment
ref_id	INTEGER	Reference where the comment is made
comment	TEXT	Comment to the reference
username	VARCHAR(255)	Username of the user who posted the comment
name	VARCHAR(255)	Name of the user who posted the comment
date_posted	DATETIME	Date when the comment is posted

Table 11: reference_comment table

tool_folder- list of tool folders of the projects in the system

Field	Type	Description
tools_folder_id	INTEGER	Unique identifier of the tool folder
tool_folder_name	VARCHAR(200)	Name of tool folder
tool_folder_desc	TEXT	Description of tool folder
tool_folder_path	VARCHAR(200)	Path where tool folder can be found
creator	VARCHAR(255)	User who created the tool folder
proj_id	INTEGER	Project where the tool folder belongs
date_created	DATETIME	Date when the folder was created

Table 12: tool_folder table

tool- list of tools of the projects in the system

Field	Type	Description
tool_id	INTEGER	Unique identifier of the tool
tool_name	VARCHAR(200)	Name of tool
tool_desc	TEXT	Description of tool
tool_link	VARCHAR(200)	Link of other information about the tool
tool_path	VARCHAR(200)	Path where tool can be found
creator	VARCHAR(255)	User who uploaded the tool
proj_id	INTEGER	Project where the tool belongs

Table 13: tool table

miscfile_folder- list of miscellaneous files folders of the projects in the system

Field	Type	Description
miscfile_folder_id	INTEGER	Unique identifier of the miscellaneous files folder
miscfile_folder_name	VARCHAR(200)	Name of miscellaneous files folder
miscfile_folder_desc	TEXT	Description of miscellaneous files folder
miscfile_folder_path	VARCHAR(200)	Path where miscellaneous files folder can be found
creator	VARCHAR(255)	User who created the miscellaneous files folder
proj_id	INTEGER	Project where the miscellaneous files folder belongs
date_created	DATETIME	Date when the folder was created

Table 14: miscfile_folder table

misc_file- list of miscellaneous files of the projects in the system

Field	Type	Description
misc_file_id	INTEGER	Unique identifier of the miscellaneous file
misc_file_name	VARCHAR(200)	Name of miscellaneous file
misc_file_desc	TEXT	Description of miscellaneous file
misc_file_link	VARCHAR(200)	Link of other information about the miscellaneous file
misc_file_path	VARCHAR(200)	Path where miscellaneous file can be found
creator	VARCHAR(255)	User who uploaded the miscellaneous file
proj_id	INTEGER	Project where the miscellaneous file belongs

Table 15: misc_file table

forum_thread- list of topics in the forum

Field	Type	Description
thread_id	INTEGER	Unique identifier of the thread
proj_id	INTEGER	Project where the thread belongs
title	TEXT	Title of the thread
creator	VARCHAR(255)	User who posted the thread
date_created	DATE	Date when the thread is created

Table 16: forum_thread table

thread_comment- list of comments to the thread

Field	Type	Description
comment_id	INTEGER	Unique identifier of the comment
thread_id	INTEGER	Forum thread where the comment is posted
comment	TEXT	Comment to the thread
username	VARCHAR(255)	Username of user who posted the comment
name	VARCHAR(255)	Name of user who posted the comment
date_posted	DATE	Date when the comment is posted

Table 17: thread_comment table

svn_repository- subversion repository account of a project

Field	Type	Description
proj_id	INTEGER	Project where the subversion account belongs
url	VARCHAR(255)	URL of the subversion repository
username	VARCHAR(255)	Username used in the subversionn repository
password	VARCHAR(100)	Password of the subversion repository

Table 18: svn_repository table

V. Architecture

A. System Architecture

CSCoRE is built using the Google Web Toolkit framework. In its presentation layer, it has the GWT user interface components which includes the HTML, CSS, and JavaScript. The JavaScript component comes from the compiled Java codes by GWT. It connects with Java servelets which utilizes the application programming interfaces (API) to communicate with the storage layer. The MySQL Query API is used to communicate with the system database and the SVNKit is the API used to connect to the subversion repository. Figure 64 shows the architecture of the system.

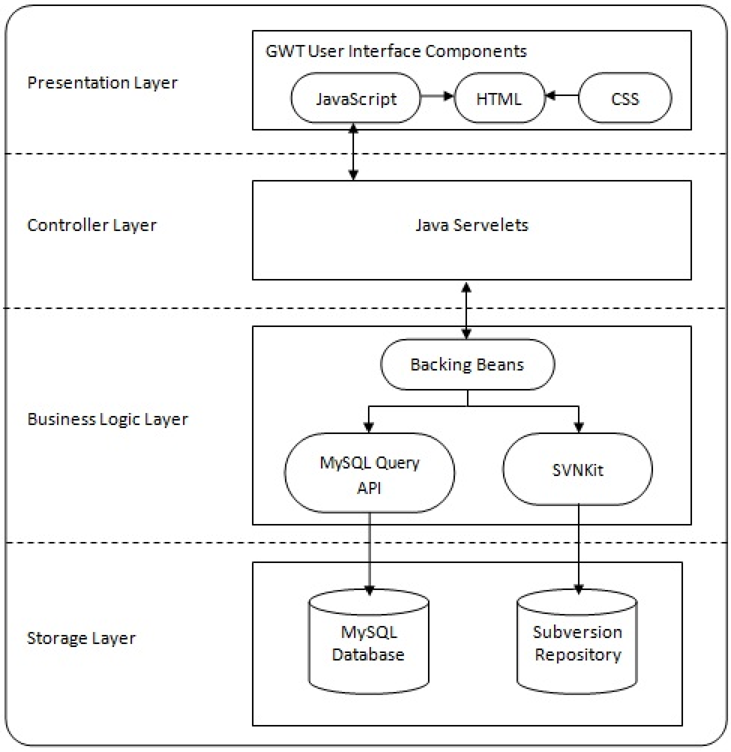


Figure 64: CSCoRE system architecture

1. Google Web Toolkit (GWT)

Google Web Toolkit is a development toolkit for building and optimizing complex browser-based applications. With GWT, the developers write the AJAX front-end in the Java programming language which GWT then cross-compiles into optimized JavaScript that automatically works across all major browsers. [22]

Projects created with GWT usually have 2 major components, the server side codes which do the database process and the client side codes which are responsible for the user interface actions. The client side component communicates with the server-side component via remote procedure calls. Figure 65 shows the GWT architecture overview.

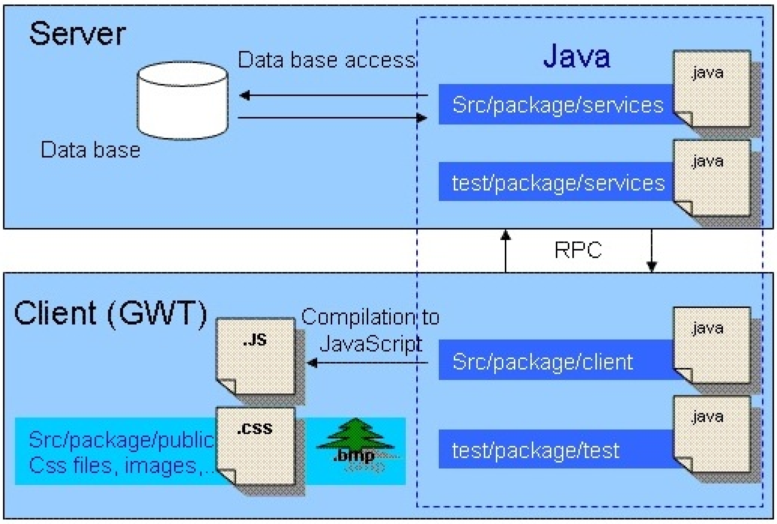


Figure 65: GWT architecture overview

2. Model View Presenter (MVP)

Similar to CEpiR, CSCoRE is also built using the Model View Presenter design. MVP is an architecture highly recommended by Google to use with GWT. In MVP, the Model encompasses the business objects of the system. The View contains all of the UI components that make up the application while the Presenter contains all the logic for the system. The fourth component, the AppController, controls all the logic outside the Presenter such as the history management and view transition logic. The Presenter catches all the events done in the View and passes the event to the AppController. Both the Presenter and AppController uses the RPC service if access to the database and subversion repository is needed. [22] Figure 66 shows the MVP architecture of CSCoRE.

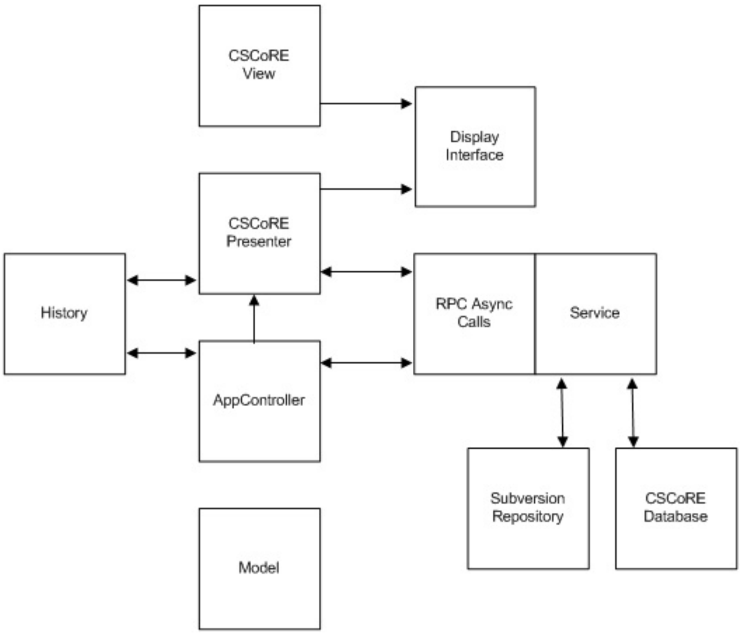


Figure 66: MVP architecture of CSCoRE

B. Technical Architecture

1. Server

Hardware Requirements

- CSCoRE requires 100MB of free disk space and at least 512MB RAM.

Software Requirements

- The system is designed to run using Apache Tomcat 6.0 and MySQL 5.

- It also requires Java Runtime Environment 1.5.

- For the operating system, CSCoRe runs on Windows XP/Vista/7, or Ubuntu.

2. Client

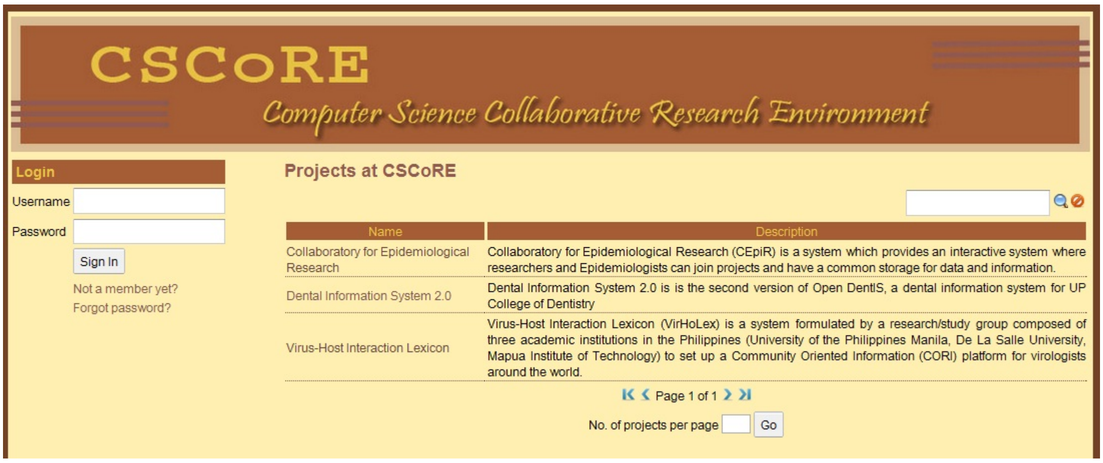
CSCoRE requires the client to have Web browser. It works on:

- Google Chrome 18.0.1025.142

- Mozilla Firefox 11.0

VI. Results

The homepage of the Computer Science Collaborative Research Environment is shown on Figure 67. The users and visitors can view the list of the projects in the system. In the home page, the system allows registered users to login and lets the guests to request for an account. Also, registered users can reset their password by clicking the Forgot Password link.

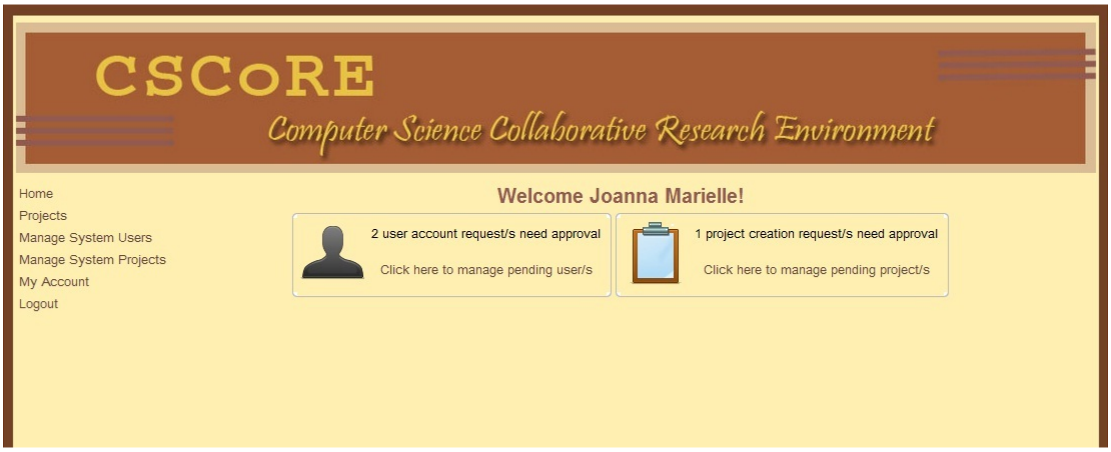


Projects at CSCoRE

Name	Description
Collaboratory for Epidemiological Research	Collaboratory for Epidemiological Research (CEpiR) is a system which provides an interactive system where researchers and Epidemiologists can join projects and have a common storage for data and information.
Dental Information System 2.0	Dental Information System 2.0 is is the second version of Open DentIS, a dental information system for UP College of Dentistry
Virus-Host Interaction Lexicon	Virus-Host Interaction Lexicon (VirHoLex) is a system formulated by a research/study group composed of three academic institutions in the Philippines (University of the Philippines Manila, De La Salle University, Mapua Institute of Technology) to set up a Community Oriented Information (CORI) platform for virologists around the world.

Figure 67: Homepage of CSCoRE

When the user logs in to the system, the he/she will be directed to the dashboard. If the user is a system administrator, the system displays the number of all pending account and project creation requests of the system. Otherwise, the system displays all the user's pending project creation requests, pending project membership requests and if applicable, all the project membership requests of sent to the projects where the user is a project administrator. Figures 68 and 69 show the homepage for system administrators and registered users, respectivively.



Welcome Joanna Marielle!

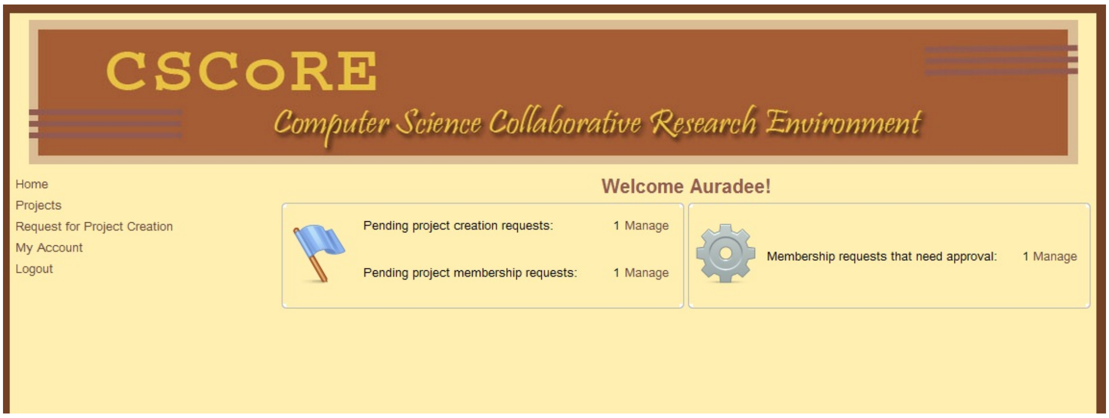
2 user account request/s need approval

Click here to manage pending user/s

1 project creation request/s need approval

Click here to manage pending project/s

Figure 68: Homepage for system administrators


CSCoRE
Computer Science Collaborative Research Environment
Home
Projects
Request for Project Creation
My Account
Logout

Welcome Auradee!

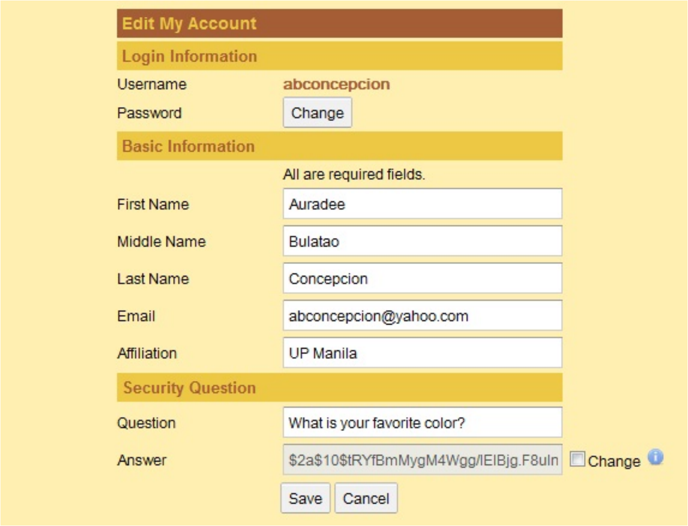
Pending project creation requests: 1 Manage

Pending project membership requests: 1 Manage

Membership requests that need approval: 1 Manage

Figure 69: Homepage for registered users

Registered users can edit their account information and change their passwords. Figure 70 shows the edit account information.



Edit My Account

Login Information

Username	abconcepcion
Password	Change

Basic Information

All are required fields.

First Name	Auradee
Middle Name	Bulatao
Last Name	Concepcion
Email	abconcepcion@yahoo.com
Affiliation	UP Manila

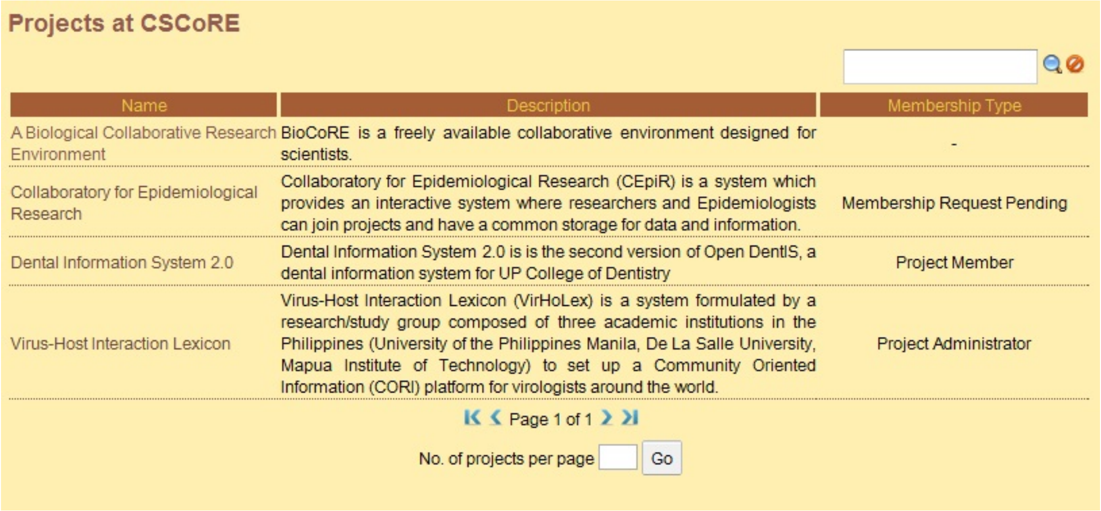
Security Question

Question	What is your favorite color?
Answer	$2a$10$tRYfBmMygM4Wgg/lEIBjg.F8uln □ Change

Save Cancel

Figure 70: Edit account information

They can also view the list of projects in the system. Indicated in the list also is the status of the membership of a user in the project. Figure 71 shows the view all projects page of the system.



Projects at CSCoRE

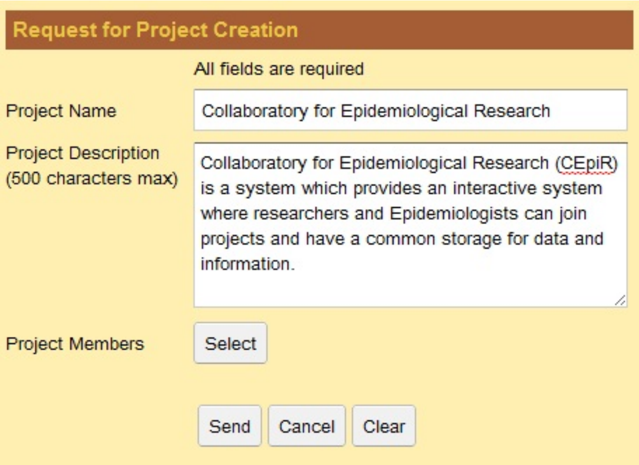
Name	Description	Membership Type
A Biological Collaborative Research Environment	BioCoRE is a freely available collaborative environment designed for scientists.	-
Collaboratory for Epidemiological Research	Collaboratory for Epidemiological Research (CEpiR) is a system which provides an interactive system where researchers and Epidemiologists can join projects and have a common storage for data and information.	Membership Request Pending
Dental Information System 2.0	Dental Information System 2.0 is is the second version of Open DentIS, a dental information system for UP College of Dentistry	Project Member
Virus-Host Interaction Lexicon	Virus-Host Interaction Lexicon (VirHoLex) is a system formulated by a research/study group composed of three academic institutions in the Philippines (University of the Philippines Manila, De La Salle University, Mapua Institute of Technology) to set up a Community Oriented Information (CORI) platform for virologists around the world.	Project Administrator

Page 1 of 1

No. of projects per page Go

Figure 71: View all projects in CSCoRE

Moreover, registered users can request for project creation. Figure 72 show the project creation request form.



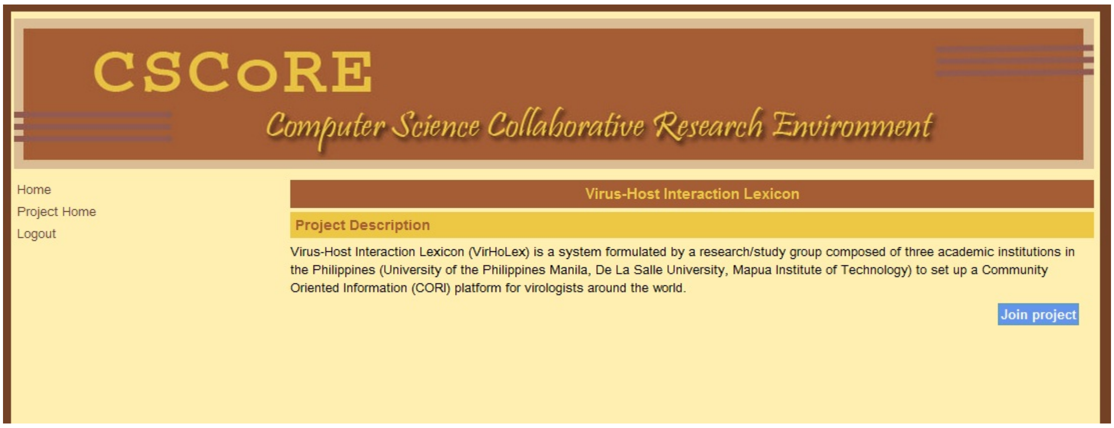
Request for Project Creation

All fields are required

Project Name	Collaboratory for Epidemiological Research
Project Description (500 characters max)	Collaboratory for Epidemiological Research (CEpiR) is a system which provides an interactive system where researchers and Epidemiologists can join projects and have a common storage for data and information.
Project Members	

Figure 72: Project creation request form

Each project in the system has its own home page. Non-members can request for membership using this page while members can use this page to leave the project. Only the project administrators have the option to edit or terminate the project. Figures 73 and 74 show the poject homepage for non-members and project administrators.

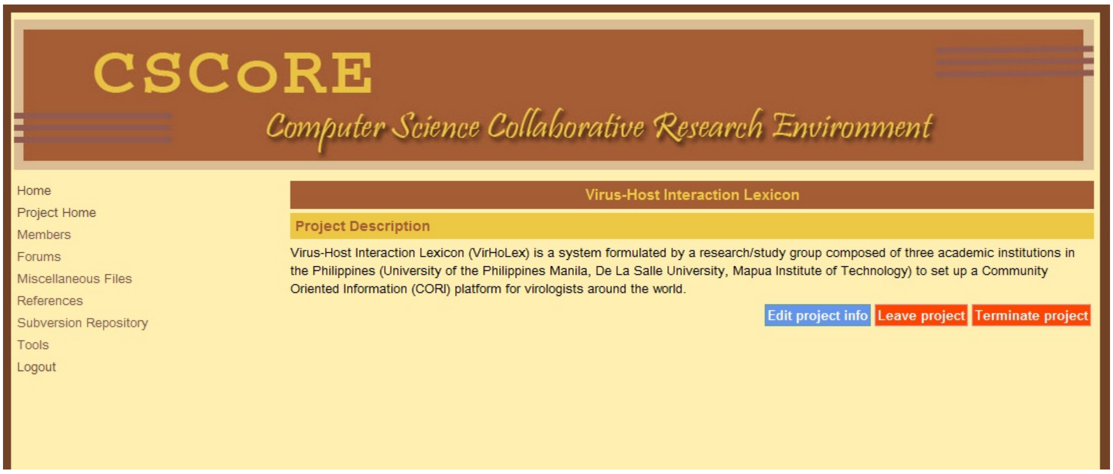


Virus-Host Interaction Lexicon

Project Description

Virus-Host Interaction Lexicon (VirHoLex) is a system formulated by a research/study group composed of three academic institutions in the Philippines (University of the Philippines Manila, De La Salle University, Mapua Institute of Technology) to set up a Community Oriented Information (CORI) platform for virologists around the world.

Figure 73: Project homepage for non-members



Virus-Host Interaction Lexicon

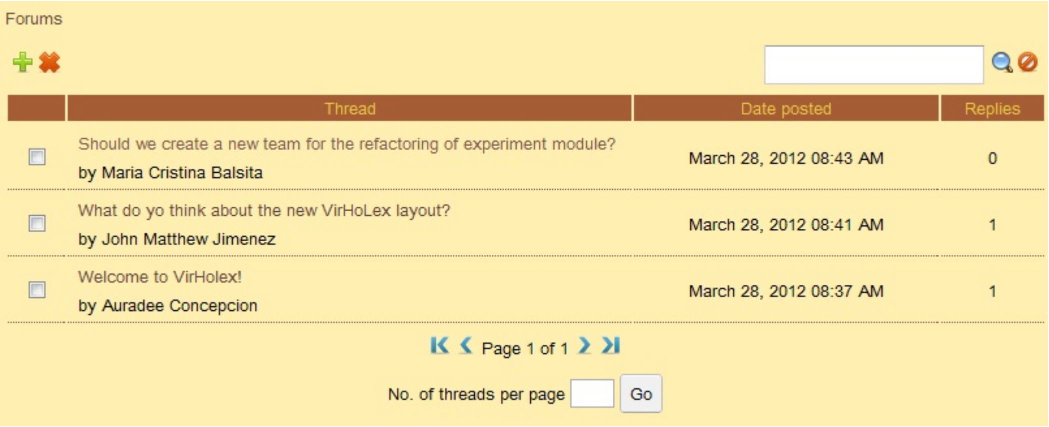
Project Description

Virus-Host Interaction Lexicon (VirHoLex) is a system formulated by a research/study group composed of three academic institutions in the Philippines (University of the Philippines Manila, De La Salle University, Mapua Institute of Technology) to set up a Community Oriented Information (CORI) platform for virologists around the world.

Figure 74: Project homepage for project administrators

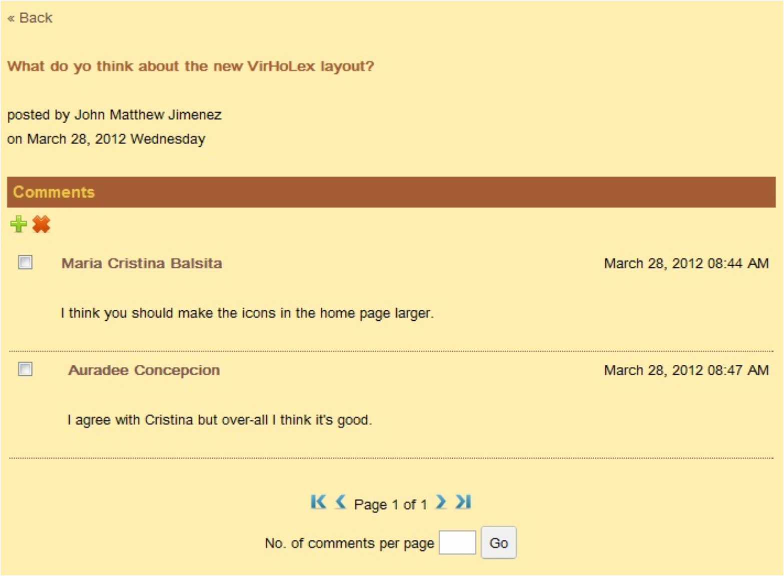
Currently, each project in CSCoRE has 5 modules: Forums, Miscellaneous Files, References, Subversion and Tools. Any member can view and download the contents of each module. However, only the project contributors and project administrators can add, edit and delete items from them.

Figure 75 shows the main page of the forums component and figure 76 shows the view forum thread page.



	Thread	Date posted	Replies
□	Should we create a new team for the refactoring of experiment module? by Maria Cristina Balsita	March 28, 2012 08:43 AM	0
□	What do yo think about the new VirHoLex layout? by John Matthew Jimenez	March 28, 2012 08:41 AM	1
□	Welcome to VirHolex! by Auradee Concepcion	March 28, 2012 08:37 AM	1

Figure 75: Main page of forums module



« Back

What do yo think about the new VirHoLex layout?

posted by John Matthew Jimenez
on March 28, 2012 Wednesday

Comments

✚ ✖

□ Maria Cristina Balsita — March 28, 2012 08:44 AM

I think you should make the icons in the home page larger.

□ Auradee Concepcion — March 28, 2012 08:47 AM

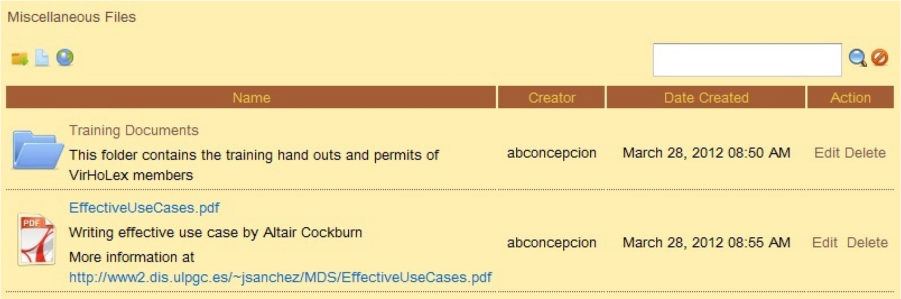
I agree with Cristina but over-all I think it's good.

⏮ ‹ Page 1 of 1 › ⏭

No. of comments per page □ Go

Figure 76: View forum thread

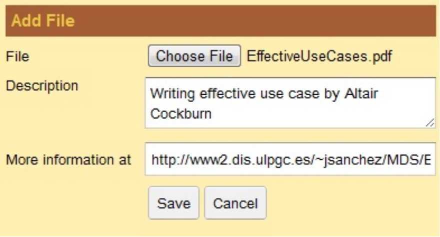
Figure 77 shows the main page of the miscellaneous files component and figure 78 shows the add miscellaneous file page.



Miscellaneous Files

Name	Creator	Date Created	Action
Training Documents This folder contains the training hand outs and permits of VirHoLex members	abconcepcion	March 28, 2012 08:50 AM	Edit Delete
EffectiveUseCases.pdf Writing effective use case by Altair Cockburn More information at http://www2.dis.ulpgc.es/~jsanchez/MDS/EffectiveUseCases.pdf	abconcepcion	March 28, 2012 08:55 AM	Edit Delete

Figure 77: Main page of miscellaneous files module

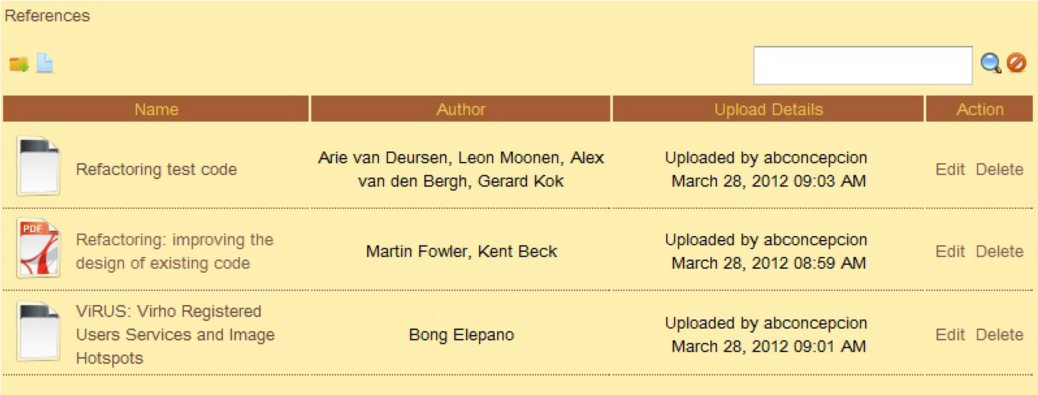


Add File

File	Choose File EffectiveUseCases.pdf
Description	Writing effective use case by Altair Cockburn
More information at	http://www2.dis.ulpgc.es/~jsanchez/MDS/E
	Save Cancel

Figure 78: Add miscellaneous file

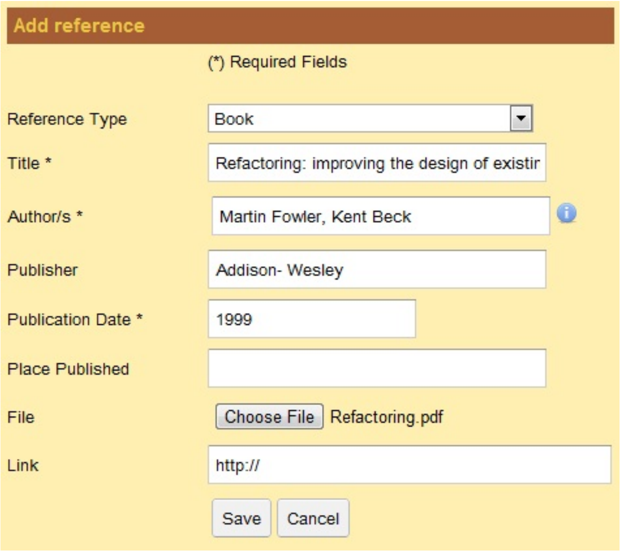
Figure 79 shows the main page of the references component, figure 80 shows the add reference page and figure 81 shows the view reference page.



References

Name	Author	Upload Details	Action
Refactoring test code	Arie van Deursen, Leon Moonen, Alex van den Bergh, Gerard Kok	Uploaded by abconcepcion March 28, 2012 09:03 AM	Edit Delete
Refactoring: improving the design of existing code	Martin Fowler, Kent Beck	Uploaded by abconcepcion March 28, 2012 08:59 AM	Edit Delete
ViRUS: Virho Registered Users Services and Image Hotspots	Bong Elepano	Uploaded by abconcepcion March 28, 2012 09:01 AM	Edit Delete

Figure 79: Main page of references module

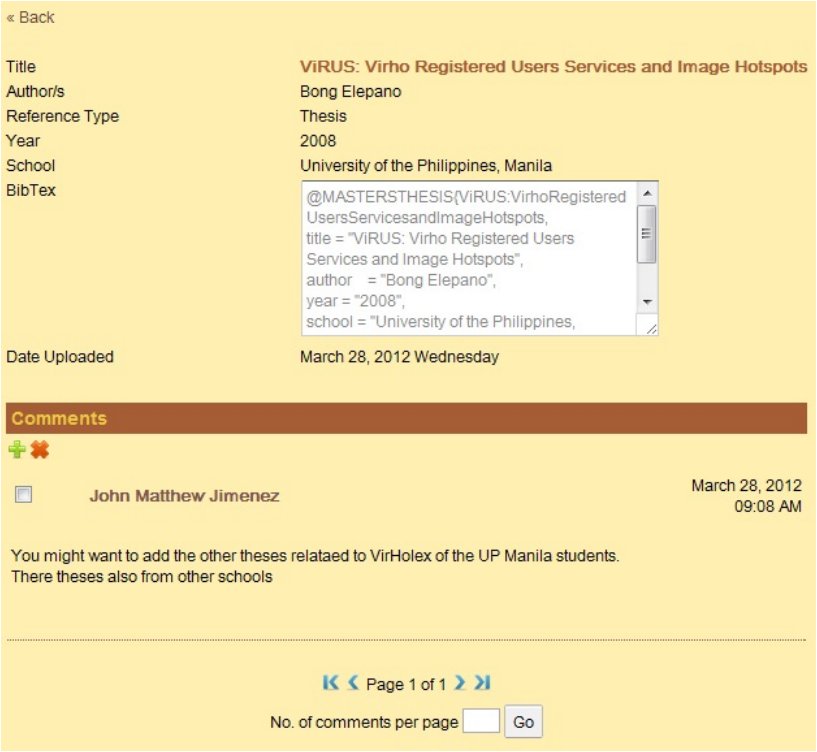


Add reference

(*) Required Fields

Reference Type	Book
Title *	Refactoring: improving the design of existir
Author/s *	Martin Fowler, Kent Beck
Publisher	Addison- Wesley
Publication Date *	1999
Place Published	
File	Choose File Refactoring.pdf
Link	http://

Figure 80: Add reference



Title	ViRUS: Virho Registered Users Services and Image Hotspots
Author/s	Bong Elepano
Reference Type	Thesis
Year	2008
School	University of the Philippines, Manila

BibTex	@MASTERSTHESIS{ViRUS:VirhoRegistered
	UsersServicesandImageHotspots,
	title = "ViRUS: Virho Registered Users
	Services and Image Hotspots",
	author = "Bong Elepano",
	year = "2008",
	school = "University of the Philippines,

Date Uploaded	March 28, 2012 Wednesday

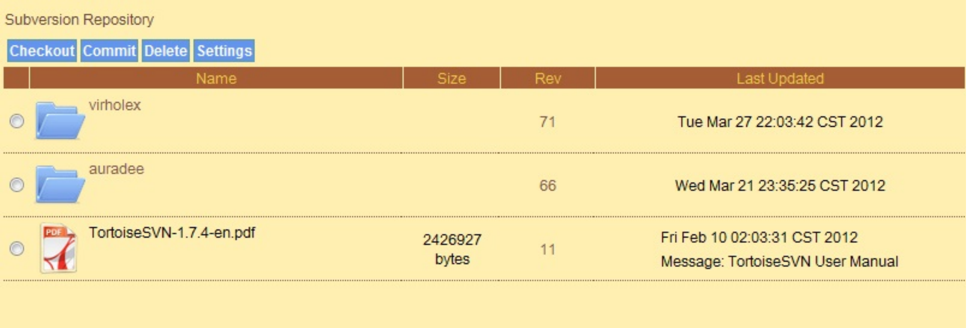
Comments

☐ John Matthew Jimenez March 28, 2012 09:08 AM

You might want to add the other theses relataed to VirHolex of the UP Manila students.
There theses also from other schools

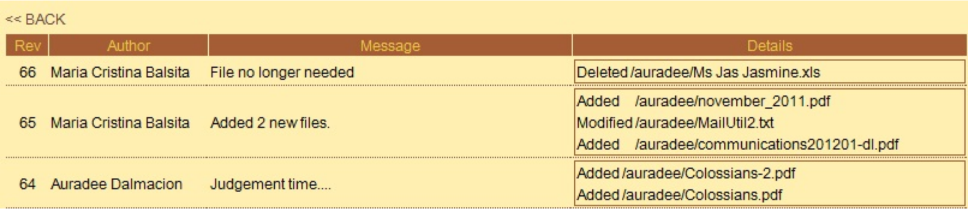
Figure 81: View reference

Figure 82 shows the main page of the subversion component and figure 83 shows the view repository history page.


Subversion Repository

	Name	Size	Rev	Last Updated
○	virholex		71	Tue Mar 27 22:03:42 CST 2012
○	auradee		66	Wed Mar 21 23:35:25 CST 2012
○	TortoiseSVN-1.7.4-en.pdf	2426927 bytes	11	Fri Feb 10 02:03:31 CST 2012 Message: TortoiseSVN User Manual

Figure 82: Main page of subversion module

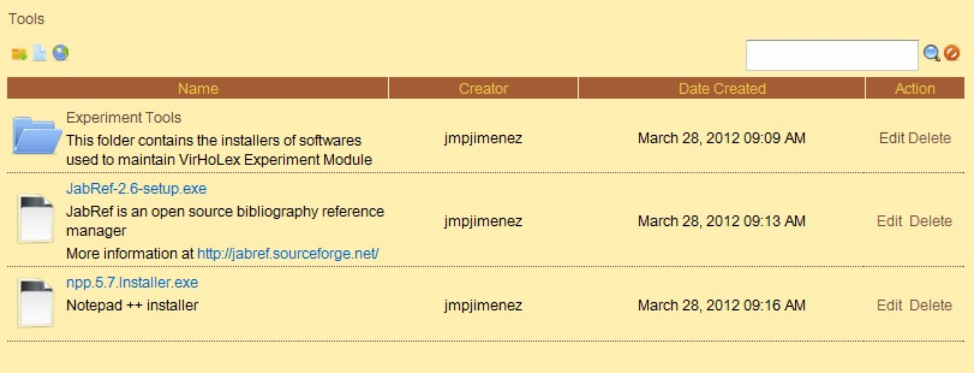


<< BACK

Rev	Author	Message	Details
66	Maria Cristina Balsita	File no longer needed	Deleted /auradee/Ms Jas Jasmine.xls
65	Maria Cristina Balsita	Added 2 new files.	Added /auradee/november_2011.pdf Modified /auradee/MailUtil2.txt Added /auradee/communications201201-dl.pdf
64	Auradee Dalmacion	Judgement time....	Added /auradee/Colossians-2.pdf Added /auradee/Colossians.pdf

Figure 83: View repository history

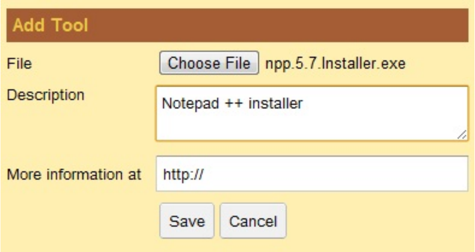
Figure 84 shows the main page of the tool component and figure 85 shows the add tool page.



Tools

Name	Creator	Date Created	Action
Experiment Tools This folder contains the installers of softwares used to maintain VirHoLex Experiment Module	jmpjimenez	March 28, 2012 09:09 AM	Edit Delete
JabRef-2.6-setup.exe JabRef is an open source bibliography reference manager More information at http://jabref.sourceforge.net/	jmpjimenez	March 28, 2012 09:13 AM	Edit Delete
npp.5.7.Installer.exe Notepad ++ installer	jmpjimenez	March 28, 2012 09:16 AM	Edit Delete

Figure 84: Main page of tools module



Add Tool

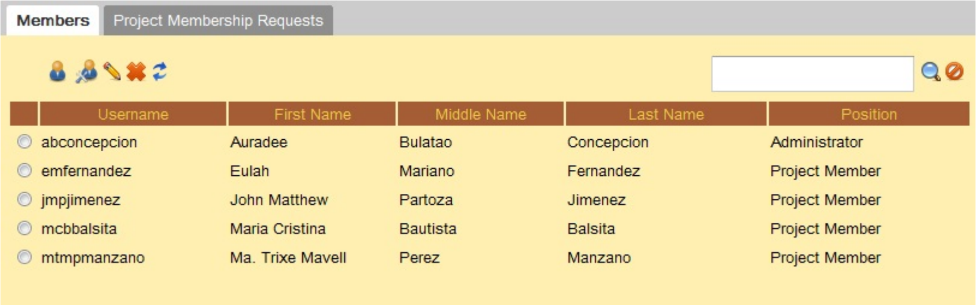
File: Choose File npp.5.7.Installer.exe

Description: Notepad ++ installer

More information at: http://

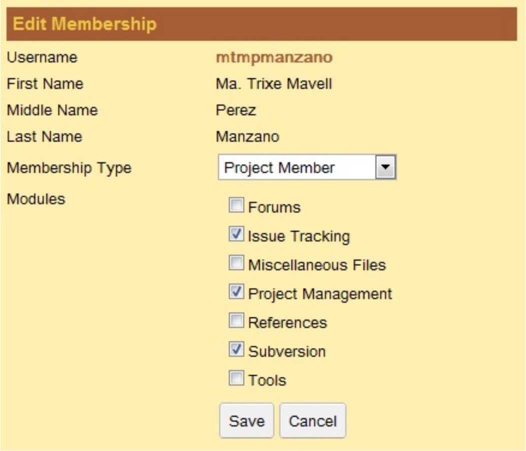
Figure 85: Add tool

If the user is a project administrator, he/she can manage members of the project. Figure 86 shows the members list of the project and the available options for the project administrator. Figure 87 shows the edit membership page where the administrator can select which module the user can peroform add, edit and delete processes.



	Username	First Name	Middle Name	Last Name	Position
○	abconcepcion	Auradee	Bulatao	Concepcion	Administrator
○	emfernandez	Eulah	Mariano	Fernandez	Project Member
○	jmpjimenez	John Matthew	Partoza	Jimenez	Project Member
○	mcbbalsita	Maria Cristina	Bautista	Balsita	Project Member
○	mtmpmanzano	Ma. Trixe Mavell	Perez	Manzano	Project Member

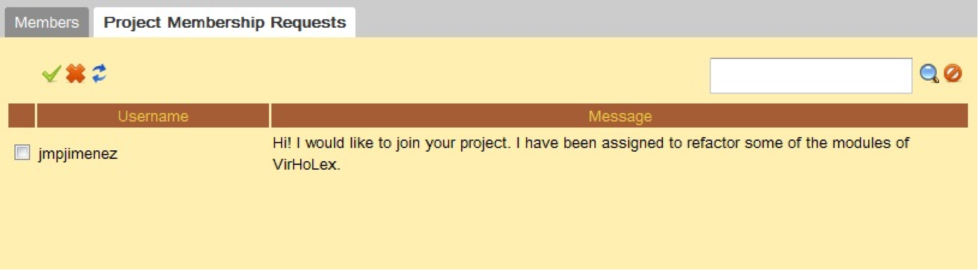
Figure 86: Manage project members page


Edit Membership

Username	mtmpmanzano
First Name	Ma. Trixe Mavell
Middle Name	Perez
Last Name	Manzano
Membership Type	Project Member
Modules	☐ Forums ☑ Issue Tracking ☐ Miscellaneous Files ☑ Project Management ☐ References ☑ Subversion ☐ Tools

Figure 87: Edit membership

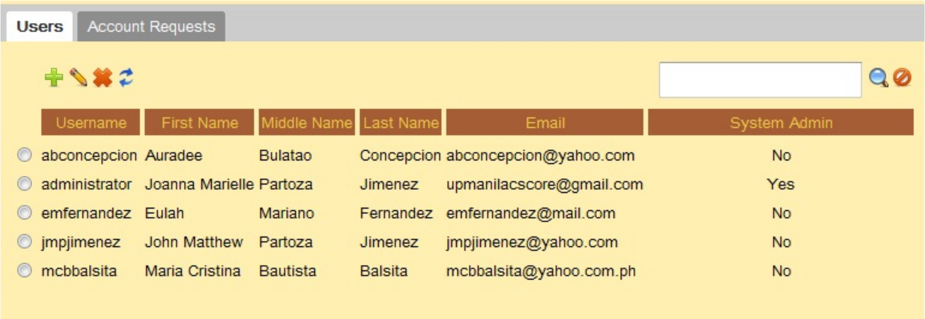
Project administrators can also approve or reject project membership requests. Users are notified via email once their request has been approved or rejected. Figure 88 shows the pending membership requests tab.



	Username	Message
☐	jmpjimenez	Hi! I would like to join your project. I have been assigned to refactor some of the modules of VirHoLex.

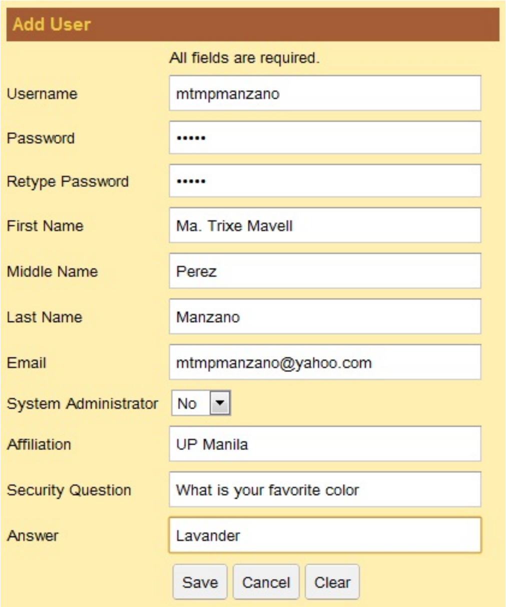
Figure 88: Pending membership requests

System administrators can add, edit and delete any user in the system, except for himself. Figure 89 shows the manage users page and Figure 90 shows the add user form.



	Username	First Name	Middle Name	Last Name	Email	System Admin
○	abconcepcion	Auradee	Bulatao	Concepcion	abconcepcion@yahoo.com	No
○	administrator	Joanna Marielle	Partoza	Jimenez	upmanilacscore@gmail.com	Yes
○	emfernandez	Eulah	Mariano	Fernandez	emfernandez@mail.com	No
○	jmpjimenez	John Matthew	Partoza	Jimenez	jmpjimenez@yahoo.com	No
○	mcbbalsita	Maria Cristina	Bautista	Balsita	mcbbalsita@yahoo.com.ph	No

Figure 89: Manage system users

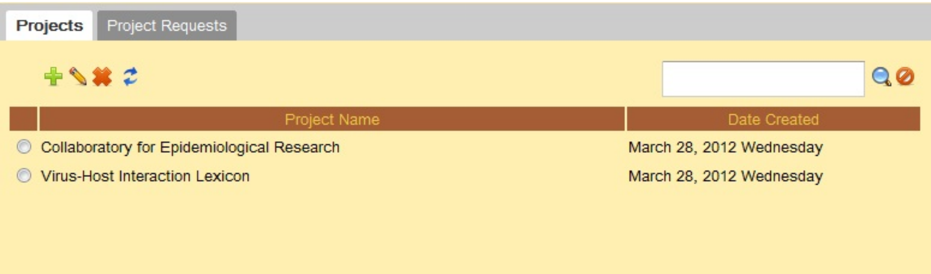


All fields are required.

Username	mtmpmanzano
Password	•••••
Retype Password	•••••
First Name	Ma. Trixe Mavell
Middle Name	Perez
Last Name	Manzano
Email	mtmpmanzano@yahoo.com
System Administrator	No
Affiliation	UP Manila
Security Question	What is your favorite color
Answer	Lavander

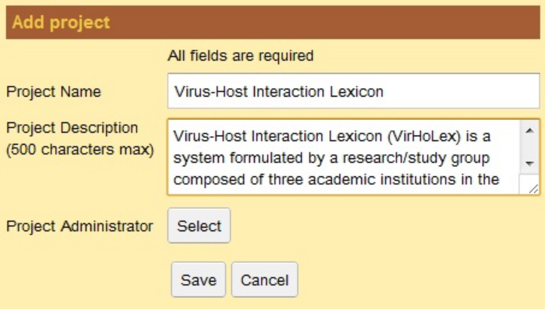
Figure 90: Add user form

System administrators can also add, edit and delete any project in the system. Figure 91 shows the manage projects page and Figure 92 shows the add project form.



	Project Name	Date Created
○	Collaboratory for Epidemiological Research	March 28, 2012 Wednesday
○	Virus-Host Interaction Lexicon	March 28, 2012 Wednesday

Figure 91: Manage system projects



All fields are required

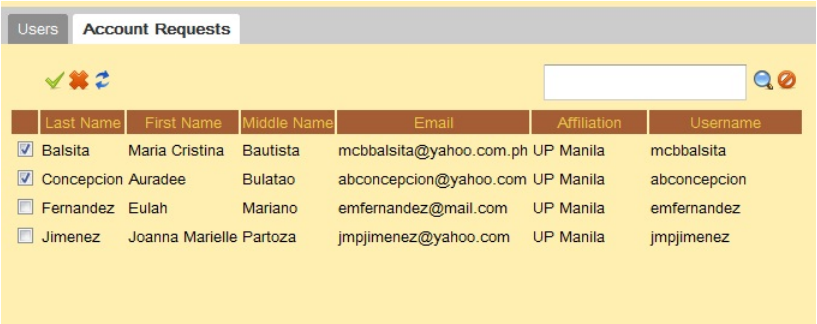
Project Name: Virus-Host Interaction Lexicon

Project Description (500 characters max): Virus-Host Interaction Lexicon (VirHoLex) is a system formulated by a research/study group composed of three academic institutions in the

Project Administrator

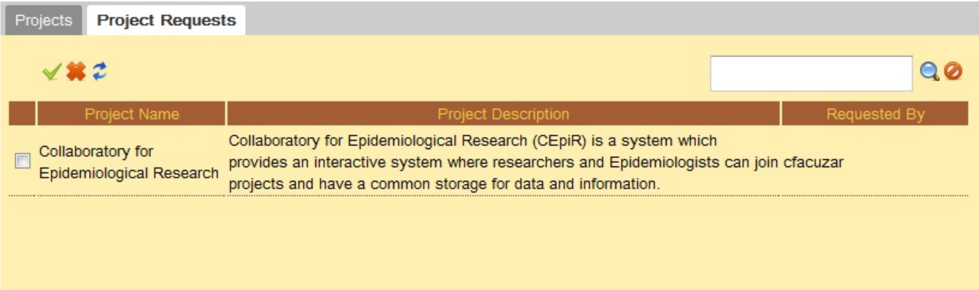
Figure 92: Add project form

Furthermore, system administrators can approve or reject account and project creation requests. Users are also notified via email if their request has been approved or rejected. Figures 93 and 94 shows the manage account and project creation request tabs.



	Last Name	First Name	Middle Name	Email	Affiliation	Username
☑	Balsita	Maria Cristina	Bautista	mcbbalsita@yahoo.com.ph	UP Manila	mcbbalsita
☑	Concepcion	Auradee	Bulatao	abconcepcion@yahoo.com	UP Manila	abconcepcion
□	Fernandez	Eulah	Mariano	emfernandez@mail.com	UP Manila	emfernandez
□	Jimenez	Joanna Marielle	Partoza	jmpjimenez@yahoo.com	UP Manila	jmpjimenez

Figure 93: Manage project creation request tab



	Project Name	Project Description	Requested By
□	Collaboratory for Epidemiological Research	Collaboratory for Epidemiological Research (CEpiR) is a system which provides an interactive system where researchers and Epidemiologists can join projects and have a common storage for data and information.	cfacuzar

Figure 94: Manage project creation request tab

VII. Discussions

The generic collaborative component of the Collaboratory for Epidempiological Research (CEpiR) has been adapted to build CSCoRE. Like CEpiR, CSCoRE is able to provide services to different types of users. In the system, the non-registered users are able to browse the list of projects. They can also request for an account. For the registered users, the system allows them to send requests for project creation and requests for project membership.

Members of the project are classified based on their access level to each module. Members who are under the restricted users level are able to view and download the items in the tools, references and miscellaneous files modules. The system also allows them to browse the threads in the forums module and check-out files from the version control module. They are also able to receive updates from the projects via e-mail. On the other hand, members who are classified as project contributors can also perform all the functions of resctricted users. In addition to these functions, the system also allows project contributors to add, edit and delete the items in the tools, references and miscellaneous files module. They can also participate and post threads and comments in the forums module. In the version control module, the system allows them to delete and commit changes to the files in the repository.

For each project, there is at least one project administrator which the system allows to manage the project and its member. Project administrators can approve or reject membership requests to his projects. He can also edit the membership of each user within the project and within the modules of the project. Furthermore, the system lets him to edit the project information and terminate the project.

For the system, there are system administrators which have the privilege to manage the user accounts and projects. They can add, edit, view and delete user accounts and projects. Also, system administrators are responsible in approving or rejecting user account and project creation requests.

Each module in CSCoRE contributes to effective collaboration. The miscellaneous files and tools components provides a common storage for handling of files and tools which gives users a quick and easy way to manage and access them. The references component is helpful in organizing information sources and research references. The forums module, facilitates group discussion about a certain topic which is an efficient way to exchange ideas to other group members. The subversion module allows the users to browse the repository and directly checkout and commit changes to files. Similar to usual repository browsers, the version control in CSCoRE allows the users to view the files and its history. But unlike other systems, CSCoRE provides a direct way to access files in the repository. It does not require the installation and use of third party clients. However, the functions available as of now are limited to just checking out, commiting changes to files, deleting and viewing repository history. The last component, the user management module, allows administrators to control the membership of project members and easily manage who can manipulate the data within each module of the project thus enabling security to the group.

VIII. Conclusions

The Computer Science Collaborative Research Environment (CSCoRE) is a tool for collaborative software development and software project management that is specifically designed for Computer Science research. It promotes effective collaboration by integrating the essential tools for such activity. With this system, users need not to have an account in different applications and keep each updated. CSCoRE has provided effective handling of documents thus providing easy and quick access to tools, references and other files. Moreover, by providing a common storage and interface, participant can engage in collaboration even if they are from different geographical location. With the use of projects, the system allows the participants to share ideas and work together privately.

One novel feature of the system is the integration of Subversion (SVN). Subversion manages files and directories, and the changes made to them, over time. The SVN module of CSCoRE, similar to usual repository browsers, allows the users to view the files and examine the history of how the data changed. But unlike others, the system allows the users to directly check-out and commit changes to files without the use of third party client.

IX. Recommendations

CSCoRE is a web based system which enables its registered users to join projects and share data among themselves. Its modules can be extended to promote more effective collaboration. The version control module can be improved by providing the other functionalities of Subversion such as retrieving previous versions of files and resolving conflicts. Furthermore, the system has only been using a single account in the repository for each of the project. In order to fully maximize the use of the version control system, it is recommended that management of user accounts in the Subversion repository should be included in the system. In the references module, a feature can be added in such a way that project members can download the BibTex file of a reference or references. Conversely, it can be also included in that component the uploading of references with details given via BibTex entry. Lastly, its forums module can be enhanced to have rich text format editor which can help emphasizing important details in the user's replies.

The system can also be improved by adding other modules such as a Calendar module. Other version control systems like Git and CVS can be included too.

X. Bibliography

[1] S. Katz and B. Martin, "What is research collaboration?," *Research Policy*, vol. 26, pp. 1–18, 1997.

[2] "Collaborative research." http://www.rcr.emich.edu/module9. Accessed on Sept. 12, 2011.

[3] T. Yamakami, "A stage model of open source activities: an exploratory analysis on open source repository," in *Proceedings of the 12th international conference on Advanced communication technology*, (Piscataway, NJ, USA), pp. 915–919, IEEE Press, 2010.

[4] C. Gutwin, R. Penner, and K. Schneider, "Group awareness in distributed software development," in *Proceedings of the 2004 ACM conference on Computer supported cooperative work*, (New York, NY, USA), pp. 72–81, ACM, 2004.

[5] S. Komi-Sirvio and M. Tihinen, "Lessons learned by participants of distributed software development," *Knowledge and Process Management*, vol. Volume 12, no. 2, p. 108122, 2005.

[6] L. Vela, "Towards a definition of collaboration," September 2010.

[7] E. Ly, "Distributed java applets for project management on the web," *IEEE Internet Computing*, vol. 1, pp. 21–26, 1997.

[8] N. Bos, A. Zimmerman, J. Olson, J. Yew, J. Yerkie, E. Dahl, and G. Olson, "From shared databases to communities of practice: A taxonomy of collaboratories," *Journal of Computer-Mediated Communication*, vol. 12, pp. 652–672, 2007.

[9] M. Bhandarkar, G. Budescu, W. F. Humphrey, J. A. Izaguirre, S. Izrailev, L. V. Kale, D. Kosztin, F. Molnar, J. C. Phillips, and K. Schulten, "Biocore: A collaboratory for structural biology," 1999.

[10] V. Sunderam, S. Y. Cheung, M. Hirsch, S. Chodrow, M. Grigni, A. Krantz, I. Rhee, P. Gray, S. Olesen, P. Hutto, and J. Sult, "Ccf: a framework for collaborative computing," *IEEE Internet Computing*, vol. 4, pp. 16–24, 2000.

[11] S. Kaur, V. Mann, V. Matossian, R. Muralidhar, and M. Parasha, "Engineering a distributed computational collaboratory," in *System Sciences, 2001. Proceedings of the 34th Annual Hawaii International Conference on*, 2001.

[12] C. Lushbough, M. Bergman, C. Lawrence, D. Jennewein, and V. Brendel, "Bioextract server: An integrated workflow-enabling system to access and analyze heterogeneous, distributed biomolecular data," *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, vol. 7, pp. 12–24, 2010.

[13] M. Fokaefs, K. Bauer, and E. Stroulia, "Wikidev 2.0: Web-based software team collaboration," in *Wikis for Software Engineering, 2009. WIKIS4SE '09. ICSE Workshop on*, pp. 67 – 77, 2009.

[14] Y. Gao, M. V. Antwerp, S. Christley, and G. Madey, "A research collaboratory for open source software research," in *First International Workshop on Emerging Trends in FLOSS Research and Development, 2007. FLOSS '07.*, pp. 4 – 4, 2007.

[15] T. N. Nguyen, E. V. Munson, J. T. Boyland, and C. Thao, "Flexible fine-grained version control for software documentsa," in *Proceedings of the 11th Asia-Pacific Software Engineering Conference*, APSEC '04, (Washington, DC, USA), pp. 212–219, IEEE Computer Society, 2004.

[16] P. Mukherjee, C. Leng, W. W. Terpstra, and A. Schurr, "Peer-to-peer based version control," in *2008 14th IEEE International Conference on Parallel and Distributed Systems*, 2008.

[17] H. E. Lam and P. Maheshwari, "Task and team management in the distributed software project management tool," in *25th Annual International*

Computer Software and Applications Conference, 2001. COMPSAC 2001., 2001.

[18] F. Lanubile, "Collaboration in distributed software development," in *Software Engineering* (A. Lucia and F. Ferrucci, eds.), ch. Collaboration in Distributed Software Development, pp. 174–193, Berlin, Heidelberg: Springer-Verlag, 2009.

[19] R. Kouzes, J. Myers, and W. Wulf, "Collaboratories: Doing science on the internet," *Computer*, vol. 29, no. 8, pp. 40–46, 1996.

[20] C. Acuzar, "Collaboratory for epidemiological research," 2011.

[21] B. Collins-Sussman, B. Fitzpatrick, and M. Pilato, "Version control with subversion."

[22] "Google web toolkit." https://developers.google.com/web-toolkit/. Accessed on March 28,2012.

XI. Appendix

A. Source Codes

XII. Acknowledgement

My whole college life seems like a roller coaster ride for me and this special problem has been the peak of that whole journey. The successful completion of this requirement isn‘t possible without the invaluable contribution of a number of people. Much has been the twists and turns, and the ups and downs I have experienced with them and to say “thank you” is not even enough to express my appreciation.

Foremost, I would like to express my sincere gratitude to the person who has helped me reach the peak of this journey, my adviser, Sir Richard Bryann Chua. He has given us, his advisees, great efforts, patience and diligence in guiding us in doing our proposals, systems and documentations. Also, he has given us a little more than enough stress and pressure which really pushed us to finish our SPs. But most importantly, he has given us that one precious thing that nobody can ever return to him, his time.

I am deeply grateful also to my family: to my Mama and Papa who always remind me that no matter what happen they will be there for me. They always make me feel that I am loved despite being a crybaby for almost 20 years. My thanks also to my brothers, Jem, Jig, and Pax, even if they‘re really naughty and noisy. But they serve as my company whenever I go home. To my Lolo and Lola, and my Mommy and Daddy thank you for your never-ending support and encouragement. I love you all so much!

It gives me great pleasure too in acknowledging the support of Carmel Acuzar, the author of CEpiR, the system in which I based mine. She‘s the one who I bug whenever I have questions about GWT and other stuffs related to our systems. I won‘t be able to start the development my SP if it isn‘t because of her. I would like to thank her for she openly shares her experiences about the making of her Special Problem.

I also would like to thank all my Computer Science instructors and professors whose teachings have been essential in the development of my system.

Now, to most awesome people of all, the members of ComSci Dose ‘08, I just can‘t find words to express my gratitude to all of you. Being with you guys and gals lets me forget for a moment all my problems and be extremely happy. Thank you for all the parties, chitchats, banters, gimmicks and eat outs which really relieve the stresses in me. I may have not mentioned you one by one but each of you has really made a great impact on my life and has able to help me somehow to reach where I am now.

To the one above all of us, the Omnipresent God, I thank You for answering my prayers and for giving me the wisdom, competence, hope, and strength to finish not only my special problem but my course as well. Thank You also for giving me the people around me who are like angels sent from above.

Well, thank you for those who prayed and believed in me. What I have written above may not be enough to really express my gratitude to everyone. The ride may be fast and short but with you all it has been great all along.